



BONUS CD-ROM! Macromedia Studio MX 2004 trial & lots more!



BOOST YOUR SKILLS WITH

FLASH

Everything you need to
create compelling content

**FOR
MAC &
PC**



**SIMPLE
STEPS
SERIES
23**

FROM THE MAKERS OF
COMPUTER

arts.net

EXCLUSIVE
TO
WHSmith







Welcome

Whether you're an experienced user or newcomer to Flash, you'll find plenty here to whet your appetite and expand your skills – on both Mac and PC!

Without doubt, this is the finest collection of software resources and information you'll find to help you unlock the full

power of Macromedia Flash MX 2004. Many publications treat Flash as a powerful and capable multimedia application by itself, but it isn't really a stand-alone program. It can be used as such, as we show in Chapter 3, but its real efficiency and power is easily demonstrated when you use it in conjunction with the other elements of Macromedia's industry-standard Studio MX suite.

When Studio MX 2004 was released, it set new precedents for cross-compatibility in creative software. Not only do the MX 2004 iterations introduce new, improved versions of the already formidable Macromedia 'big five' programs – Flash, Fireworks, FreeHand, ColdFusion and Dreamweaver – but all these are now better integrated than ever, sharing a number of similar features and interface conventions, promising faster, more seamless workflow. Find out how to use the other elements of Studio MX 2004 in conjunction with Flash in Chapter 2.

Flash development used to be restricted: dial-up modems limited the audience for large file sizes and rich content, and people

There's now huge scope for specially targetted Flash content for 3G mobile phones and other portable gadgets

were bound to their computers for viewing Flash files. This is no longer true: wide take-up of broadband has meant an exciting explosion in more sophisticated Flash creativity, and with around 50% of mobile phones sold in 2005 being 3G-capable – not to mention many portable gaming devices and PDAs – there's now huge scope for producing specially targetted Flash and Shockwave content for the on-the-move audience.

Even the most experienced users are finding the wealth of new features and options available to them in Flash MX 2004 to be a boon. For instance, Timeline Effects has reduced the need for excessive keyframing; PDF and EPS files are supported, the new History panel makes producing macros a breeze, and the built-in spellchecker and reusable Publish settings take much of the boring, repetitive work out of your tasks. Furthermore, runtime performance is boosted with the enhanced compiler, working

with databases is straightforward with new levels of Microsoft Access compatibility, international content is managed using great Globalization tools, and ActionScript 2.0 will be appreciated especially by programmers with Java experience. Turn to Chapter 4 to get some great Tricks of the Trade from our multimedia professionals.

Your cover-mounted CD-ROM is packed with fantastic software. As well as a 30-day trial version of Studio MX 2004, it contains all the files you need to follow our tutorials, some great full and trial software.

Turning your inspiration into a finished project has never been easier – with the help of *Boost Your Skills With Flash*, creativity is just a few clicks away!

Matt



Media with passion

Operations Editor Matt Nailon
digitalbookazines@futurenet.co.uk
Art Editor Andy Ounsted
Production Editor Jon Palmer
Publisher Kelley Corten

Published by Future Publishing Limited
30 Monmouth Street, Bath BA1 2BW United Kingdom
www.futurenet.co.uk www.myfavouritemagazines.co.uk
Entire contents © 2005 Future Publishing Ltd. All rights reserved

Subscriptions and back issues

If you have a query regarding a subscription or back issue, or would like to place an order, please contact our customer services team:
UK Telephone: 0870 8374722 Overseas Telephone: +44 (0)1858 438794
Email: customerservice@futurenet.co.uk

Overseas Licensing Enquiries

International Licensing Director Simon Wear
simon.wear@futurenet.co.uk +44 (0)1225 822798

Future Publishing is part of Future plc

Future plc produces carefully targeted specialist magazines for people who share a passion. We aim to satisfy that passion by creating titles offering value for money, reliable information, smart buying advice and which are a pleasure to read. Today we publish more than 100 magazines in the UK, US, France and Italy. Over 100 international editions of our magazines are also published in 30 other countries across the world. Future plc is a public company quoted on the London Stock Exchange (symbol: FUTR).

Non-executive Chairman
Chief Executive
Managing Director
Group Finance Director

Roger Parry
Greg Ingham
Robert Price
John Bowman

Tel +44 (0)1225 442244 www.futureplc.com
Bath • London • Milan • New York • Paris • San Francisco

CHAPTER 1

Getting started with Web design

Even if you've already got some experience with Flash and the other elements of Macromedia Studio MX 2004, there's plenty of know-how here that's guaranteed to revolutionise the way you work...

10	PROFILE: Coolest jobs on the net
14	Web design for peanuts
17	First steps in Web design
24	Inspirational new design techniques
32	Designing for Broadband
38	Royalty-free images
40	Is film dead?
42	Web design nightmares
46	Make your site standard
51	Building websites for everyone
56	Accessibility and compliance

CHAPTER 2

Studio MX 2004 projects

The beauty of Macromedia's Studio MX suite lies its consistency. This chapter shows you how Flash interoperates with other elements such as Fireworks and Dreamweaver in a complimentary way...

62	PROFILE: Zoolook
68	PROFILE: Hi-Res
70	Building games with Studio MX 2004

74	Create games in Flash
80	Shoot-'em-ups
84	Understanding the Dreamweaver interface
86	CSS-based layers
90	Virtual Tour
94	Photo Album
98	Flash Guestbook
102	Contribute
106	Protect your work

CHAPTER 3

Flash MX 2004 techniques

There's no better way to gain new skills and inspiration than through practical experience – these tutorials show you that Flash finesse is just a few clicks away...

112	SHOWCASE: Flash
116	PROFILE: Electrotank
118	Flash techniques
120	Building Flash components
126	Information management
130	Storyboarding a movie
134	Rotoscoping video
139	Make an icon font
143	Type in motion
148	Typographic animation
152	Better characters
156	Character animation tips
160	Lip and facial synching
164	Countdown Clock
168	Sliding Puzzle

CHAPTER 4

Web design tips and tricks

Successful finishing touches require attention to detail. Discover the secrets of the web professionals as they expose their favourite techniques and shortcuts...

174	Return of the code
178	Optimise images for the web
184	50 ways to better web graphics
188	Graphics – Fireworks
190	Tricks of the Trade part 1
194	Tricks of the Trade part 2
198	Tricks of the Trade part 3
201	Tricks of the Trade part 4
203	60 ActionScript Tips

CHAPTER 5

Disc pages

A selection of the applications that accompany Macromedia Studio MX 2004 on your cover disc...

212	Flash Decompiler Lite
214	Flash Designer
216	Powerbullet Presenter
218	Comixware 2.1 trial: storyboard a short movie

ON YOUR PACKED CD

Essential software

- Macromedia Studio MX 2004 trial for Mac & PC!
- Flash Decompiler Lite **FULL PROGRAM**
- NVU – open-source web design **FULL PROGRAM**
- Comixware 2.1 trial: crazy computer cartooning
- Powerbullet Presenter – ditch PowerPoint today!
- All the files you need to follow our tutorials
- Flash Designer and lots more besides!

BOOST YOUR SKILLS WITH
FLASH

Macromedia Studio MX 2004
trial for Mac & PC

Includes:

- Dreamweaver MX 2004
- Fireworks MX 2004
- Flash MX 2004
- FreeHand MX
- ColdFusion MX (PC only)

PLUS: Tutorial files, Flash tools and lots more!

FROM THE MAKERS OF
arts.net

Boost your creativity and create compelling content



ADSLGuide.org.uk
Your Guide to Broadband Internet

Always on, no restrictions and free connection.

Welcome to ADSLGuide.org.uk!
Find the latest and greatest broadband news in the UK, service provider information, views and comments from users like you. Looking for a new broadband ISP? Check out our ISP List and ISP Comparator. Want an answer to a question? Try our popular Message Forums.

Broadband News [View Archive...]

Freedom2Surf reduces price of SDSL *Author: MrSalmon*
The price of SDSL products is slowly dropping, and Freedom2Surf are the latest to lower their prices. A big hurdle has been removed since activation is now free, normally the set up fee is around £250 to £450. Prices quoted are all excluding VAT.

Product	Contention Ratio	Price per month
SDSL 1Mbps	5:1	£180
SDSL 1Mbps	1:1	£250
SDSL 2Mbps	5:1	£250
SDSL 2Mbps	1:1	£333.34

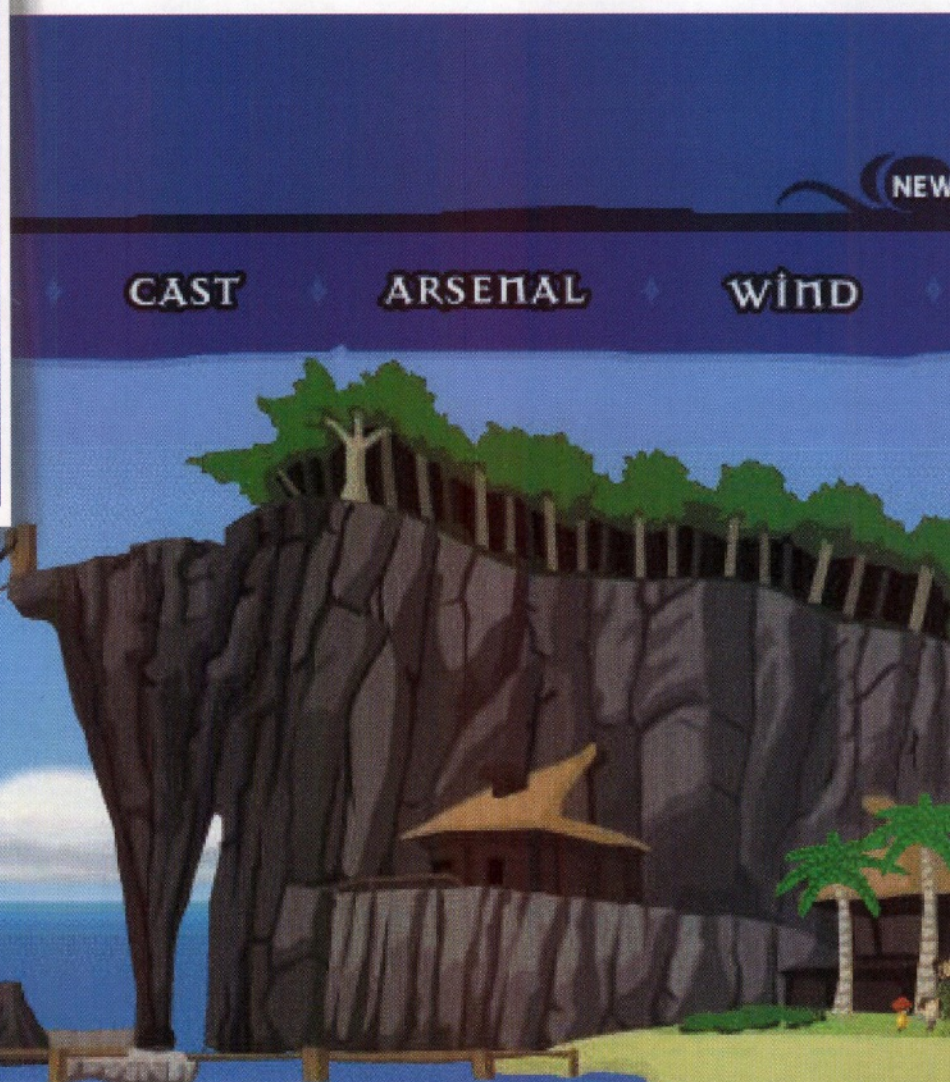
All the listed products feature a 12 month contract. For further information and contact details, see the Freedom2Surf website.
Posted: Friday, 26 January 2005, 11:53 [Link to this news item](#)

PlusNet puts users on fair usage policy *Author: MrSalmon*
PlusNet is carrying out a consultation phase with its customers over a fair usage policy for its broadband services (the products are not included). To actually consult the customers, rather than just implement a set of limits blindly is fairly unusual. If you are a PlusNet ADSL customer and this is the first you have heard of this, then we would urge you to read their statement here and fill in the survey. We must emphasise that the policy is not in place yet, PlusNet are in the phase of gathering customer feedback.

To summarise the proposed policy it will be based around a fair usage level that is set for each type of product and measured across a whole month and each time they exceed this amount a counter will increase. If during a month you do not exceed the amount the amount will be removed. Only if you

broadband BUSINESS
£18
per month
Free migration
One month contract for new installations
TOYANET SOLUTIONS
(downloading info)

News by Email
Email Address



BT YAHOO!
Broadband

Information about: Broadband | Dial-up

[Home](#) [Internet Paradise](#) [Features](#) [Compare Services](#) [Questions](#) [Easy Installation](#) [Sign Up](#) [Track Order](#)

Welcome to our Internet Paradise

BT and Yahoo! bring you BT Yahoo! Broadband

BT are leading communication experts. Yahoo! is famous for bringing you the best of the web. We teamed up to create an Internet Paradise

- Complete security
- Versatile communication
- Exclusive content
- Enhanced customisation

Come and explore
Internet Paradise



Extended January SALE! Sign up before 16th February for seasonal savings >>

Sign up online
now for no
set-up costs

Free activation
Free broadband router
Only if you sign-up online.
£25 if you order by phone.
[Full details](#)

Sign up now >>



PERSONAL
COMPUTER
WORLD

BT YAHOO!

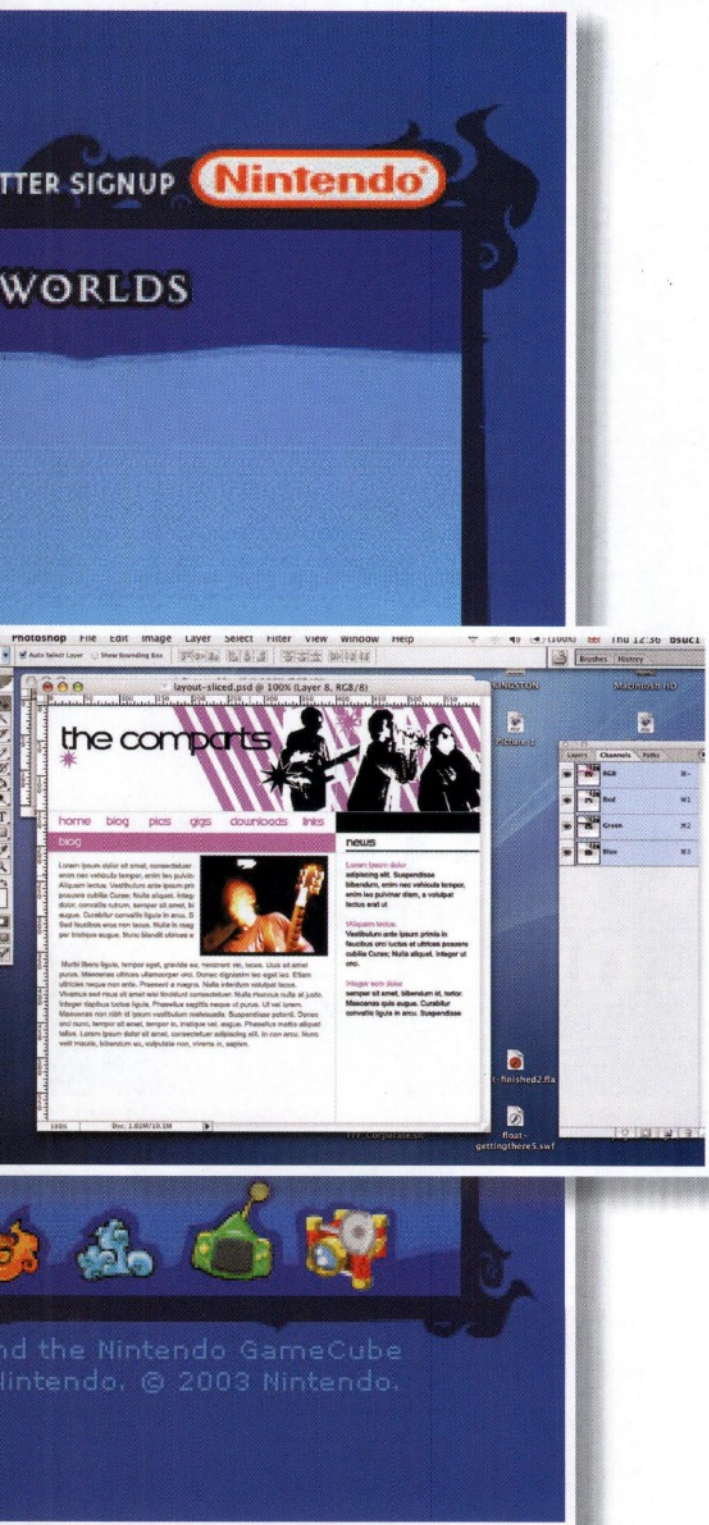
Broadband

BEST BROADBAND PROVIDER

© 2002, 2003 Nintendo, TM, logo are trademarks



1



Getting started with Web design

Even if you've already got some experience with Flash and the other elements of Macromedia Studio MX 2004, there's plenty of know-how here that's guaranteed to revolutionise the way you work...

- | | |
|----|-------------------------------------|
| 10 | PROFILE: Coolest jobs on the net |
| 14 | Web design for peanuts |
| 17 | First steps in Web design |
| 24 | Inspirational new design techniques |
| 32 | Designing for Broadband |
| 38 | Royalty-free images |
| 40 | Is film dead? |
| 42 | Web design nightmares |
| 46 | Make your site standard |
| 51 | Building websites for everyone |
| 56 | Accessibility and compliance |



The coolest jobs on the net

The web has opened up a whole new range of careers and has made it possible to turn a hobby into a job. **Oliver Lindberg** went out among the web workers to track down nine people whose unique and exciting occupations wouldn't exist without the net.

illustration: warren holder

The internet radio DJ



Mark Collymore (aka The Funky Moose)
DJ, head of music and programming and director
www.globalgrooving.com

Although nobody at online radio station Global Grooving gets paid at present (any profits are reinvested within the company), Mark has fulfilled his dream via the internet. When a friend, web designer Ian Stephenson, came up with the idea of setting up an internet radio station in November 2002, he recruited Mark and 14 other childhood friends from Croydon. "The thirty-somethings aimed to set up a station that would cater to their musical tastes and create a bridge to music-loving friends and relations based overseas," Mark recalls. "The popularity of the station has grown via word-of-mouth across the internet and it now receives well over 100,000 hits per month

from over 46 countries." Now Mark, by day a communications concept design manager and by night a DJ on Musical Freedom (one of Global Grooving's most popular shows), recruits new DJs and organises station club nights. "I also get to mingle with people who make records and, as a failed musician, I find hanging out with, or even just playing an up-and-coming artist's demo, very rewarding. The best thing that happened is seeing how record companies who weren't interested in us at the start of the station have ended up chasing us so they can give us their support. On a personal note, as a DJ it has been through Global Grooving that I could DJ at the 2004 Notting Hill carnival and receive fan mail from the Americas and the Far East!"





Motor sport mogul



Bryn Williams
Managing director of Crash Ltd
www.crash.net

Crash.net is one of the UK's leading motor sport sites, covering more than 20 championships around the world and attracting more than one million users a month. Bryn Williams has been to over 300 Grand Prix in his career and still travels to half of the big races across the world to take pictures of the cars, the drivers and the pit babes.

"I never even dreamt that there was a job going round the world taking pictures of racing cars," Bryn remembers. "If I had known that there was such a job, I would have dreamt of doing it but I never thought that it existed. I was going to events and taking pictures through the fences, thinking 'surely nobody gets paid to do this'".

Crash Ltd now consists of internet development, an online magazine, a photographic division and even a radio station. "The main thing I like about the job is the freedom. If I want to work hard, I can work hard. If I want to take it easy, I can take it easy. The travel is fantastic and I still enjoy taking photos. I see my photographs being used on the internet or in calendars and books and magazines. It's a combination of a lot of things but to be honest motor sport is my passion. It still excites me. The biggest buzz and the biggest kick, though, is when people within the motor sport industry say they look at Crash and think it's fantastic."



The internet visionary

Tsering Gyaltsen
Entrepreneur and Nepalese
cyber café owner
www.linkingeverest.com

Tsering Gyaltsen is the grandson of the only surviving Sherpa to have accompanied Sir Edmund Hillary on the first ascent of Mount Everest. His company, Linking Everest, runs an internet café in remote Nepalese village, Namche Bazaar — at an altitude of 3,400

metres. The café, complete with wireless access and high-speed satellite connection, is part of Tsering's vision to preserve Sherpa culture and to teach computer literacy to children. In 2003 Tsering managed to build the world's highest cyber café at Everest Base Camp. It operated for six weeks at 5,400 metres. "It was a one off," he said of the project, "a trial run for all this far more

important work with schools and students and helping to keep Sherpa culture intact. We bring the web to distant places so they can project themselves, benefit from the exposure and maybe young people will stay at home and be proud of being Sherpas, rather than running to Kathmandu or America." The publicity the café has attracted added to the proceeds it took, helping Tsering to get further funding for his school projects, which he is much more passionate about. The internet connection in Namche is relayed to a nearby school with several bridges and antennas. Students can use the internet to learn with a broader curriculum or to do distance learning. ➤





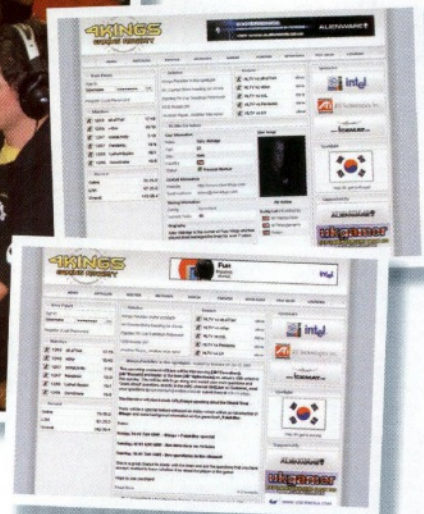
The professional gamer



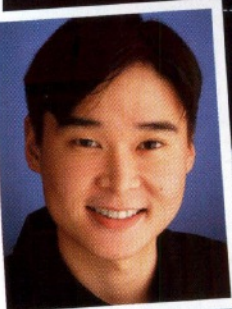
Toby Aldridge
Owner and manager of
online gaming clan, Four Kings
www.four-kings.com

Toby Aldridge manages online gaming clan, Four Kings, full-time. His duties include keeping people happy, talking to sponsors and — who'd have thought — playing games like Lineage, Painkiller and Return to Castle Wolfenstein. "I go to all of the major events," Toby says. "I've been to pretty much everywhere apart from Asia. Seeing the world at 22 is pretty cool." The money isn't bad either: an eight-week

trip to South Korea makes the Four Kings \$100,000 (about £53,562) richer. Over there, gaming is as big as football over here. "It's crazy," Toby says. "You get mugged at the airport, have to give autographs. People have posters on their wall of their favourite players. It's absolutely crazy." Sponsorship deals with Intel and other hi-tech companies ensure the 12 professionals at Four Kings get new hardware, custom-built computer cases and cash for travelling to tournaments and expenses. You need talent to get that far, though. "It's not really something that you can work on," Toby explains. "Obviously, a lot of work needs to be done but you need very good coordination and reflexes. You got to be able to think clearly under pressure. A lot of it is routine but a lot of it is natural personality and talent. Apart from that, if you've got the talent, it's practice, practice, practice."



The Google doodler



Dennis Hwang
International webmaster at Google
and graphic artist
www.google.com

Dennis Hwang is responsible for transforming the logos on Google's home pages for special occasions such as Einstein's birthday (check out www.google.com/holidaylogos.html). So far he has created more than 150 doodles but his job involves so much more. "I write code for web programming projects and help maintain our international web

pages," he says. "Drawing the holiday logos is the most visible work I do, yet my primary responsibilities are diverse. I love the working environment — the people here are brilliant, and the job allows me to work on creative projects as well as technical ones. I'm never bored at Google. And to think that millions of people around the world look at my work is both motivating and humbling."

Dennis joined Google as an intern in 2000. "When the people in the company learned that I was a studio art major in college, they gave me a chance to polish up some logos," Dennis says. "Soon I gained the founders' trust, and I've been drawing them ever since. The most challenging ones are the international logos. I spend as much time as I can on the research phase, to learn about the occasion. I read online

material and do lots of image searches to find the right symbols to represent the day. Often I read emails sent by our users for suggestions. And of course, I get help from my fellow Googlers, a vast number of which come from different countries." If you've got an idea for a doodle, simply write to doodle@google.com.



The professional hacker



Marc Maiffret
Chief hacking officer at
eEye Digital Security
www.eeye.com

Many hackers dream of turning their hobby into a career. This is exactly what Marc Maiffret, co-founder of vulnerability management software developer, eEye Digital Security, has done.

"I started like most other hacker-

types, I think," he remembers. "I got a computer one day and was consumed by it. I wanted to learn everything about how the system worked. This is something that I've felt since I was a child: always wanting to take things apart to see how they work. It just so happened that I had a knack for taking things apart and making them work in a way that they shouldn't. The next logic step was to take this hobby and attempt to make a living doing what I love. Which is what eEye is."

Marc's role as chief hacking officer is very diverse. He is responsible for everything from sales to engineering and marketing. "One day I could be helping sales close our next big deal, the next day I could be working on solving technology problems, like phishing," he explains. "The next day I could be helping launch a new version of our product, or I could be testifying before the US congress. My true passion, though, is research. The research team is the only team that I directly oversee."

Part of eEye's culture is to allow people to be themselves. This is why people at eEye get paid to do the fun stuff they would be doing even if they weren't employed. "Probably the best thing about my job is that I have seen people that I hired really excel at what they do and become great security researchers and engineers," Marc explains. "eEye is a different place to work because we are a lot more open than anywhere else."



The promoter of tomorrow's internet



Ann Doyle
Program manager for arts
and humanities for Internet2
www.internet2.edu

It's a real privilege to be able to work with tomorrow's internet. The US-based Internet2 is ultra-fast and sends data at about ten gigabits per second around the country, thus enabling multicasting — the transfer of huge data files and HDTV streaming. Internet2 is a not-for-profit project led by universities and dedicated to educational and research and development

purposes. "My job involves working with the performing arts and humanities faculty and professionals from campuses and institutions across the nation to help them understand and use advanced networking for distance education and multi-site performance events," says Ann Doyle. "I spend much of my time helping Internet2 members and colleagues understand bandwidth requirements and the equipment best suited to realise their project goals — many of which are wonderful new ideas that have never been previously tried." For Ann it's simply thrilling to see projects come to fruition that would be absolutely unthinkable with the conventional internet. "Low latency, high-quality video and audio enable performers to collaborate, perform and teach over Internet2 in ways they are unable to do over the commodity internet," she explains. Recently, she helped produce an event in which students from six campuses across the nation performed, simultaneously, for a live audience in Los Angeles. "It's fantastic," she explains. "I have been in the performing arts and information technology for 20 years now and I have truly never seen anything as extraordinary as the presence you can create over Internet2."



The live concert webcaster



Dave Meehan
Content producer
at Nyquest DVD
Productions

If you're involved in webcasting live concerts, you can not only watch the entire gig and get paid for it — often you also get the chance to meet your idols. Dave

Meehan of multimedia production company, Nyquest DVD Productions, makes all arrangements with the artists and venue, co-ordinates the camera crew and then directs the shoot. "Meeting Paul Weller and Noel Gallagher at the Steve Marriott tribute was a buzz, especially walking on stage with them," he remembers. "I also met John Lydon when we were due to webcast the Sex Pistols live at Crystal Palace. Although it fell through, he was still a

The inventive internet enthusiast



Mike Harris
Co-director and co-founder
of technical consultancy Psand
www.psand.net

Psand.net specialises in satellite and wireless communications networks for events and outreach projects. For instance, the company has come up with the iTrike, the

world's first solar-powered, wireless internet rickshaw, and a solar, wind and even pedal-powered internet café, which they take to music festivals. Psand also provided media streaming at the anti-war protests in London's Hyde Park. Mike Harris loves the diversity of his job. "I get to investigate and play with new technologies, which is what interests me about it," he explains. "I also get a chance from time to time, as with the iTrike, to apply technology in an interesting social or artistic environment and see the reaction it has on people when it's taken out of the context that they are used to — consumer devices and boring desktop computing. I also get involved in a lots of different things, from systems administration and web development, to audio-video installations and streaming."

Mike's vision is to share knowledge. He founded Psand to promote the use of open source operating systems and accompanying applications. "Ideas are a creative process between those in Psand and our collaborators," he explains. "I'm into the GNU philosophy and working with people who bring their particular skills to the fray. The most interesting thing is applying technology and seeing the results. Non-technical people tend to do the most unexpected things with technology."



great bloke to meet." Dave studied television production for three years and trained at Granada. Originally, he wanted to be a producer of a music programme but when he didn't get the chance to do that, he started webcasting gigs with the crew he already worked with. These days, Dave mostly produces live music DVDs but the process is similar. "You really need good equipment for DVD and webcasting, especially with difficult concert lighting situations," Dave explains. "Then you need to remember, whatever you are doing, the end result is going to be compressed into its suitable delivery media — whether that's Windows Media, QuickTime or in the case of DVDs Mpeg2. However, webcasting, especially for live music, isn't used half as much as it should be now broadband is popular."



Web design for peanuts



Some of the programs mentioned in this feature can be found on the cover disc.

Can't afford Macromedia's monster? Don't despair – with just £50 you can get an all-singing, all-dancing software suite and still pocket some change. **Gary Marshall** gets tooled up

For all of its joys, Dreamweaver has one big handicap – its price. At a hefty £339, it's beyond the budget of many designers; even the educational version will cost you £104 if you can get it. However, if you want the power of Dreamweaver but not the expensive price tag, there are plenty of programs to choose from and, by careful selection, you can get most of Dreamweaver's extremely useful features for less than £50 through a combination of shareware and open source software.

We've assembled two software suites –

one for PC, one for Mac – to prove that power needn't be pricey and you'll find some of them all on this month's cover disc.

To replace Dreamweaver, we think you'll need three key applications: an HTML editor, a CSS editor and an FTP program. Of course, you could just write the HTML or CSS by hand in Notepad but we think you'd be missing out. Our chosen programs make it easy to see what's happening with your code, they can help you complete tags, and they can also check your files for browser incompatibilities and standards compliance. Of course, everybody's different – we like

working in WYSIWYG mode but you may prefer to work directly in code view; some people like the combination of code and tags, while others find it annoying; most of us use PCs or Macs but you may prefer Linux. There's also some overlap between different programs: For example, some HTML programs have basic FTP features, or include a CSS editor. There are plenty of alternatives to each of our alternatives all with their own particular benefits. So, if you fancy having a go at creating your own Franken-Dreamweaver, make sure you get in touch, with your suggestions, on the forums. ■

HTML editors

Whether it's a simple home page or a complex site, a decent HTML editor is crucial

NVU

www.nvu.com

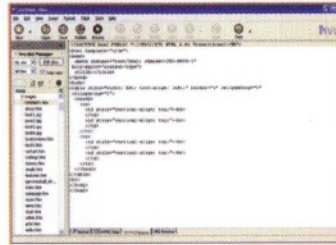
Free (open source)

NVU is a new entrant to the world of HTML editing; based on the Mozilla Editor, it's an attempt to bring decent WYSIWYG editing to Linux. It's also available for Windows and while it's still in beta format, it's surprisingly powerful with built-in FTP, template creation, a handy tabbed interface and powerful table tools. You can choose between WYSIWYG, tagged and code views to create content, and the program also includes a CSS editor, HTML validator and a JavaScript debugger. It's already a very good design package but future versions could be something very special indeed.

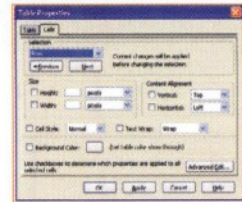
Platform: Windows, Linux

Verdict: It may be a beta, but NVU is impressive stuff. It's on the cover disc – give it a try!

FOR PC



NVU gives you four ways to look at your code: normal WYSIWYG view, an HTML tag view, the HTML source and a browser preview



One of NVU's strengths is in its table handling, which has some Dreamweaver-esque features: for example, you can right-click over table cells to merge them

PAGESPINNER

www.optima-system.com/pagespinner

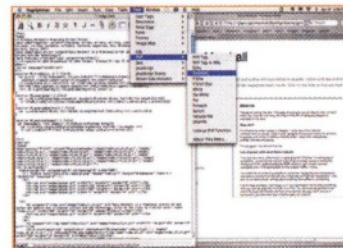
\$29.95 (£16)

This code-based HTML editor has become incredibly popular, with good reason: it's very powerful, supports PHP, JavaScript, HTML, XHTML and PHP, and can preview your pages in eight different browsers. You get customisable syntax colouring, an excellent Assistant to help you create pages, simple table creation and good CSS support, and while it isn't the easiest program to start using (at least, it isn't if you're used to WYSIWYG editing) it's well worth spending time with. You can also customise it with AppleScript to automate common tasks. It's a serious package for serious designers.

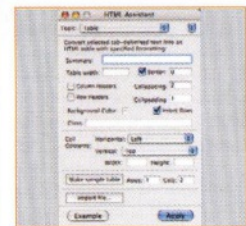
Platform: Mac OS X

Verdict: An excellent program for hand-coders.

FOR MAC



PageSpinner's menus are packed with useful features. For example the Tags menu contains tags for CSS, PHP, Java, JavaScript and Server Side Includes



PageSpinner doesn't offer WYSIWYG editing, but features such as the HTML assistant and the text menus make hand-coding as simple as possible

Save on software

Can you get big-budget software for a knock-down price?

You can save a packet by buying second-hand software from the small ads or eBay, but is it legal? The answer from Sarah Mowatt, a Macromedia spokesperson, is a cautious yes. "It is legal for a person to resell their copy of Dreamweaver, as long as they have completely removed the program from their computer," she says. "It is also legal to purchase a copy of Dreamweaver from a non-authorised vendor, but the difference is that Macromedia cannot be held responsible for the merchandise which is received."

That doesn't mean that all private sales are legal, though. As Macromedia points out: "A recent software industry 'purchase test' found that up to 90 per cent of all software purchased through non-authorised vendors (ie eBay and small ads) is illegal." Our own eBay survey confirms this, with plenty of vendors offering stuff that's clearly illegal. "One of the reasons Macromedia sells its products through authorised resellers is so it can ensure the quality and authenticity of the product," says Sarah, in response to the volume of illegal sales. If you're a student, a preferable option to potentially illegal software might be to buy the educational version (£104 at Dabs.com instead of the usual £339); however, you can't use the software for any commercial activities.

Dreamweaver is widely available on eBay, but many sellers are offering download links. Even the boxed copies are often educational versions, which can't be used for commercial web sites



Dreamweaver MX and Fireworks MX. A copy of the original software titles on one CD-ROM. The ID Key Nos. will be provided.

These software packages are a must have in all web design, and are available here at a very low price!

Features:

Dreamweaver MX

- Build Web sites and Internet applications
- Integrated workspace shared with Flash MX and Fireworks MX
- Write code faster than ever before
- Site setup wizard helps configure site information instantly
- Get started quickly with pre-built layouts and code

Fireworks MX

- Create interactive graphics in a single, Web-centric environment
- Make everything from graphical buttons to rollover effects and pop-up menus
- Open, edit, and export Photoshop graphics
- Integrate with editors like Dreamweaver and Microsoft FrontPage
- Support for major file formats and HTML standards



Another alternative

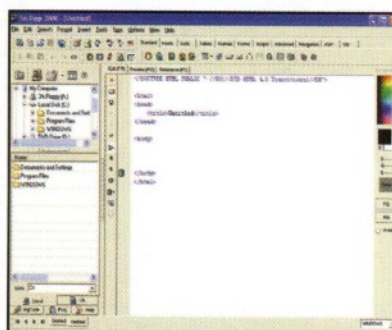
The free 1st Page 2000 is a handy HTML editor for beginners and power users alike

Download

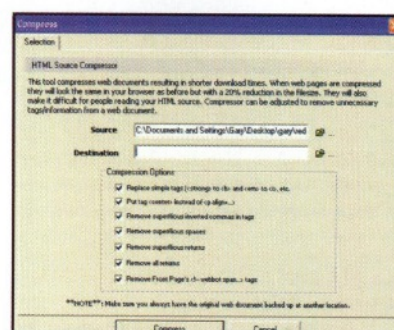
You can download 1st Page 2000 from www.evrsoft.com



1 Install 1st Page 2000. When you first run it, you'll see the Start Manager, which gives you a range of options including a blank document, a standard HTML document or a CSS document. There's also a Teach me HTML button for beginners.



2 The program doesn't offer WYSIWYG editing but it's simple to use, and you can use the tabs to switch between views at any time. Some toolbars are hidden by default; our screenshot shows the normal view but there's also a Hardcore mode that adds extra toolbar options.



3 Once you've created your page, 1st Page 2000 can make it better. HTML Tidy is integrated into the program, and you also get a handy HTML code compressor that can get rid of extra tags, line breaks and spaces to make your code as slim as possible.

CSS Editors

Use the power of CSS to get complete control over your page designs and behaviour

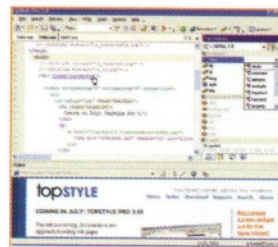
TOPSTYLE PRO
www.bradsoft.com
\$79.95 (£44)

FOR PC

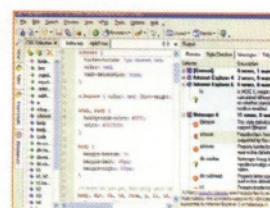
TopStyle is written by the man who created HomeSite, and it's an incredibly powerful package. The program enables you to edit HTML, XHTML and CSS in the same application, can upgrade outdated HTML to modern CSS, checks syntax against multiple browsers and can preview your pages side by side in IE and Mozilla. It integrates with external validators including the W3C validator and the Bobby Accessibility Checker, and the program can also generate reports detailing site-wide CSS usage. TopStyle also integrates with HTML Tidy, enabling you to upgrade HTML code to XHTML with a single click.

Platform: Windows

Verdict: Easy to use and amazingly powerful — but with a modest price tag.



TopStyle's user interface has good syntax highlighting and a useful Preview panel



The program includes browser error checking, and it also integrates with external validators such as the W3C Validator and the Bobby accessibility checker

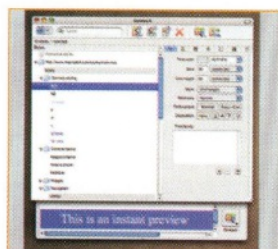
CSSEDIT
www.macrabbit.com
\$24.99 (£14)

FOR MAC

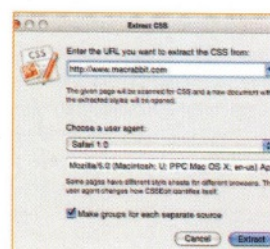
If you want to work with CSS on your Mac, then CSSEdit is definitely worth a look. The gorgeous interface includes some powerful features, including CSS extraction so you can see how other designers created their state-of-the-art sites. You can save style elements as favourites, group styles by comments, preview your style sheets in a range of different ways and take advantage of the program's intelligent auto-complete feature. The HTML linking feature is particularly clever, as it enables you to add your stylesheet to any HTML page by dragging and dropping the page into a dialogue box.

Platform: Mac OS X

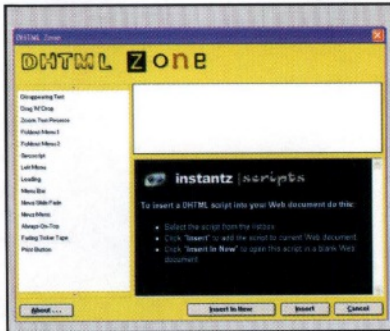
Verdict: Good looks, blistering performance and some very nifty features for less than £15.



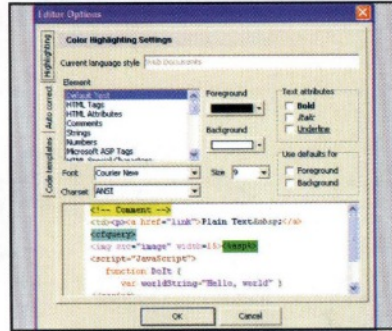
CSSEdit's interface has everything available within a few mouse clicks and it's simple to navigate between different parts of your stylesheet



Our favourite feature is the Extract CSS option, which can download sites' CSS code so you can unlock their designers' secrets



4 The program comes with a range of predefined scripts including JavaScript and DHTML. The DHTML Zone lets you see what each effect will look like, and you can then add the effect to your page or to a blank Web page with a single mouse click.



5 It's easy to customise the way in which 1st Page 2000 displays different kinds of code. You can change the font, font size, text and background colours, and the program also enables you to add or edit AutoCorrect entries to fix common typing mistakes.

Receipt

PC	
NVU:	free
TopStyle Pro:	£44
FileZilla:	free
TOTAL:	£44
Mac	
PageSpinner:	£16
CSSEdit:	£14
Transmit:	£14
TOTAL:	£44

FTP clients

Tame your transmissions and dominate downloads with a top-class FTP program

FILEZILLA

Filezilla.sourceforge.net

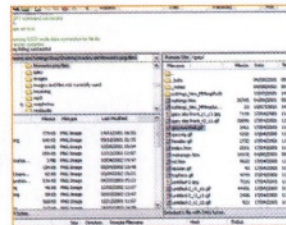
Free (open source)

Once again it's time to thank the open source community for creating decent software and giving it away for free. FileZilla is a standard FTP package that takes care of all the essentials: anonymous and secure FTP, SFTP and FTP over SSL, queued transfers and easy dragging and dropping between local and remote servers. You can transfer up to ten files simultaneously, the program includes a decent bookmark manager and it's easy to configure for proxy servers and firewalls. It isn't the most beautiful program we've ever seen but it's perfectly competent and a worthwhile addition to your Web toolkit.

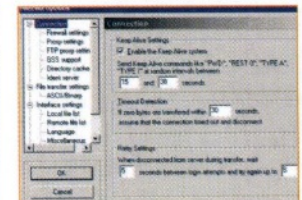
Platform: Windows

Verdict: A fast, flexible and free program with support for secure FTP

FOR PC



It isn't pretty, but FileZilla is a capable FTP client. It's exceptionally easy to use and its multi-pane interface makes site organisation simple



FileZilla's options menu provides access to more advanced features including settings for proxy servers and firewalls, connection keep-alive settings and other configuration options

TRANSMIT

www.panic.com

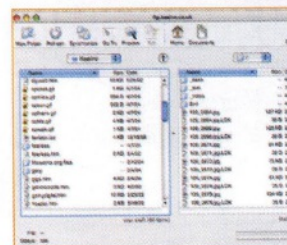
\$24.95 (£14)

Panic's excellent FTP package offers plenty of power features including batch downloading, folder synchronisation, batch downloading (from multiple sites), support for SFTP and SSH, decent management of favourite sites and support for AppleScript and Rendezvous. It also gives you quick previewing of text and image files on local and remote computers, together with a nifty option to edit text files without leaving Transmit. Like Panic's other applications it's a doddle to get to grips with, comes with a reasonable price tag and is a pleasure to use.

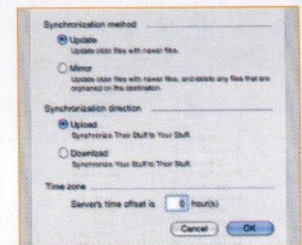
Platform: Mac OS X

Verdict: Transmit isn't just a pretty face: it's a very powerful FTP package

FOR MAC



Transmit makes the most of Mac OS X to deliver a user interface that's crystal clear without being too simplistic for power users – or too scary for beginners



The site synchronisation feature is worth its weight in gold, as it enables you to update entire web sites (or local folders) with a single click.



Taking the first steps in web design



Illustration by Danny Franzreb [w] www.taobot.com



Use Dreamweaver MX 2004 to design and build a web site for the infamous Computer Arts Projects band, The Compartments... Rock on!

Now that Macromedia Dreamweaver is in its 2004 series, things are a lot easier for those starting out in web design. With a variety of new features being added recently, building a web site can be achieved with the most limited knowledge of HTML and programming.

In this tutorial, we show you how to use Dreamweaver to create a contemporary band web site which combines both Flash and HTML elements to give it a rich look and feel. The combination of these elements enables you to create a site which is more dynamic than straight HTML but whose file size isn't so large that it takes an eternity to download.

With Dreamweaver's structured layout views, it becomes easy to learn the HTML basics because you can see Dreamweaver inserting the code behind the scenes while you work with the visual elements.

This tutorial is aimed at Dreamweaver beginners, and leads you through the processes involved in creating a web site for a fictitious band. We talk you through creating a mock-up in Photoshop, optimising images, using Dreamweaver's site management tools, importing and using imagery and Flash .swf files into Dreamweaver, as well as formatting text.



Expertise provided by Tim Ketterer and Stephen Lockhart of Float Design at www.floatdesign.net



All the components needed to complete this tutorial can be found on the cover disc.

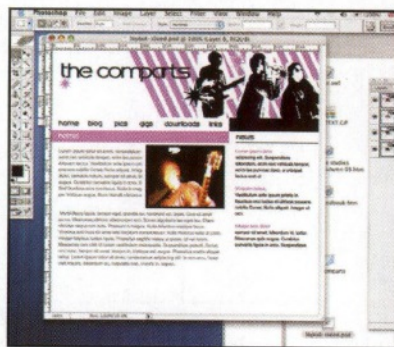


Part 1: Getting started

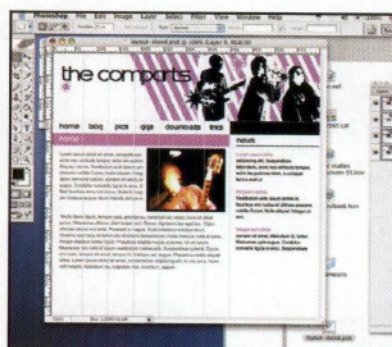
Use Photoshop and Flash to create the site imagery...

Multiple images

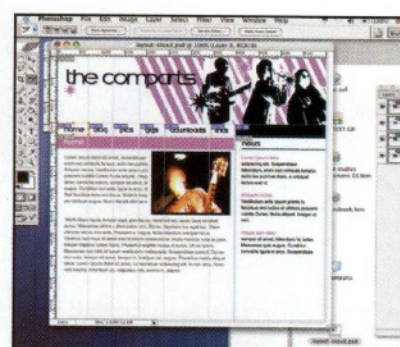
If you want to use other images within your page, use the Slice tool and name them appropriately in the Slice Options menu.



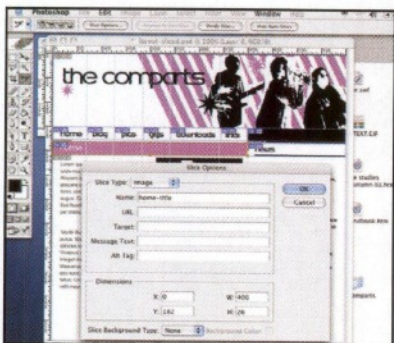
1 Create a folder on your machine, called Comparts. Mock up the site in Photoshop making sure that the design and its elements are the exact pixel size of the site. We've set the width of the document to 600px.



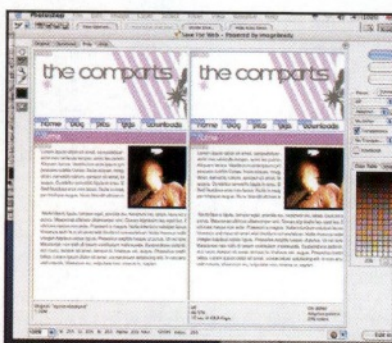
2 Use guides to split up the main areas. Think in terms of rows and columns, because each section will be placed into a table in Dreamweaver. Remember that each navigation element has to be held in its own separate column. Try to keep everything as simple as possible.



3 Use the Slice tool to slice the navigation buttons and the black bar into separate elements. Select each slice using the Slice Select tool. In the Slice Options menu, name each slice by its navigational name – for example, home, blog and so on. Ensure that the height and x-axis are the same for each slice.



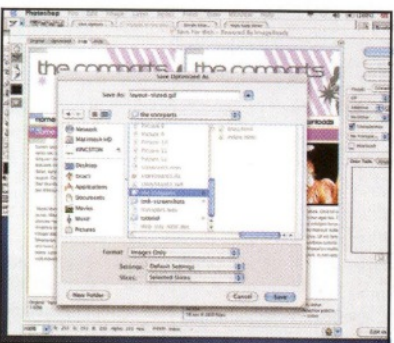
4 Next, slice the pink home image and the news image into two separate images. Name these 'home-title' and 'news' respectively. Again, check that their height and x-axis are the same.



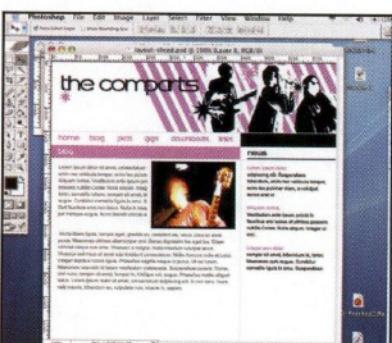
5 In the main toolbar go to File > Save for Web. In the Save for Web window click on the 2-Up tab. This shows you the original image and an optimised image. Select the Slice tool and click on one of your navigation slices.



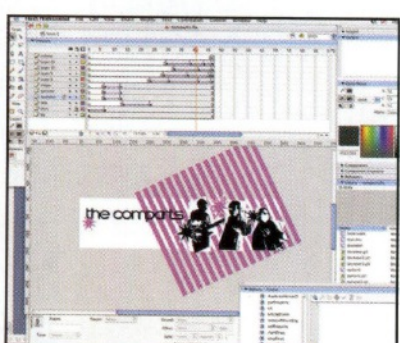
6 From the options on the right, select GIF and 256 colours. Note the file size, under the image. (Experiment with different numbers of colours and types of compression – the fewer colours, the smaller the file size. Do this for all the slices you made.)



7 Still using the Slice tool, hold Shift and select the sliced elements you want. Save. Select the folder from Step 1. Under Format, choose Images Only; and under Slices, choose Selected Slices. Save. Inside the Comparts folder Photoshop will create an images folder containing your GIF images.



8 Go back to your mock-up site and change the colour of the navigation text from black to pink. Select the slices and rename each one by appending '-roll' to the name. Save these new images for web as before.



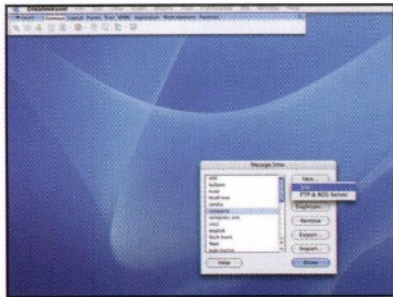
9 Open Flash MX and set the document size to 600px by 150px. Create a retro Flash animation of the band and its title font. Save the swf as comparts.swf. This will be set as the header for each page.

Part 2: Setting up Dreamweaver

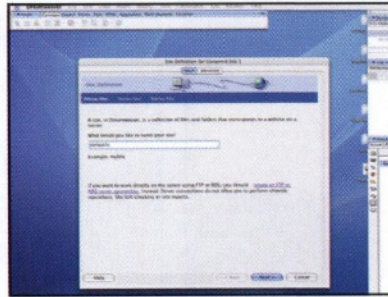
Get to grips with Dreamweaver's site management tools...

Site definition

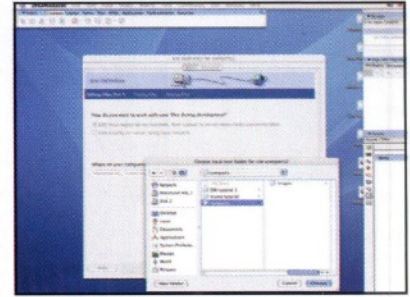
Defining your site is important because it enables Dreamweaver to know where your site exists on your machine, which in turn enables it to insert the correct paths to images and links within your HTML pages.



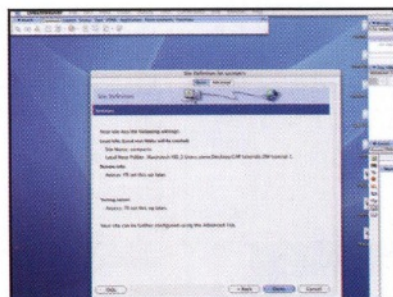
1 Open Dreamweaver MX 2004. The first thing you need to do is define your site. From the menu bar choose Site > Manage Sites... From the subsequent dialogue box select New... then choose Site from the two options.



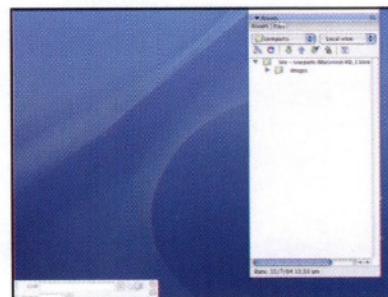
2 In the resulting Site Definition window select the Basic option at the top. In the first field, give your new site a name (for example, Comparts). Click Next, and then select No to using any server technologies. Again, click Next.



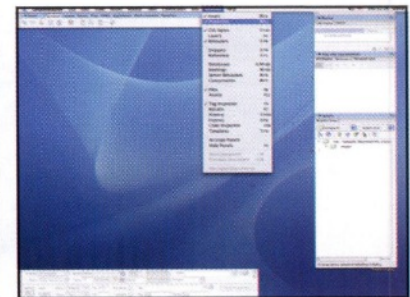
3 Select the first option: 'Edit local copies on my machine.' Then click on the folder icon to locate the folder called Comparts that you made when creating your Photoshop images. This will be the root folder for your web site, where all your HTML files will be saved. Click Next.



4 Select None for the option 'How do you connect to a remote server', and again click Next. You are now shown a summary of the options you've selected. If these are all correct, click Done, and Done again from the small Manage Sites box.



5 Make sure your Files panel is open – Window > Files (F8) – and you should see the various files and folders contained in your site (your root folder). For now, it should show the images folder that contains your Photoshop images and the Flash swf file.



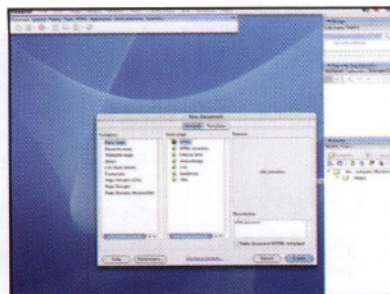
6 Dreamweaver has many panels, located under the Window menu option. Make sure you have the following panels open: the Insert panel, the Property Inspector (Properties), and the Behaviours panel (the Files panel should already be open). You'll be using all of these throughout this tutorial.

Part 3: Creating a new page

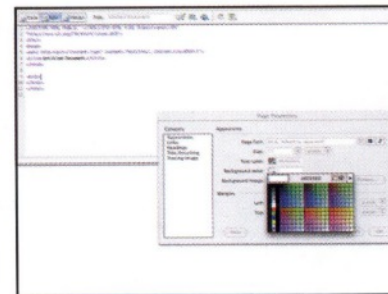
Open a new document and set its page properties...

Get the specs right

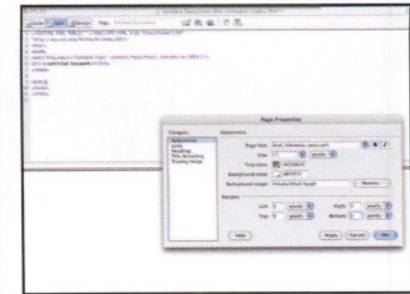
When designing your web site, you have to think of your intended user and how they will be using it. If you want your site to appeal to everybody, then you should design it for the minimum specifications so everyone can view it. As over 50 per cent of users still use an 800x600 screen resolution, you should design to this size. Therefore the maximum size of your HTML document (to be seen without any scrollbars on an 800x600 screen) should be 745px by 420px. This takes into account the various toolbars on browsers, on both Mac and PC.



1 Select File > New from the main toolbar. In the new document window, make sure General, Basic page and HTML are selected. Press Create. A new document will be created. Save this in the Comparts folder as index.html.



2 To set the properties, choose Modify > Page Properties. Make sure Appearance is selected on the left. Using the menu, set Page font to Arial, Helvetica, sans-serif, Size to 11 pixels, Text colour to #666666 (dark grey), and Background colour to #ffffff (white).



3 We've set a background image for the document – a black/grey striped image that is only 5 pixels in height but 1200 pixels wide. This will repeat to fill the page. To attach this, press Browse and navigate to black-bg.gif. Set all the margin fields to 0, and then click Apply and OK.

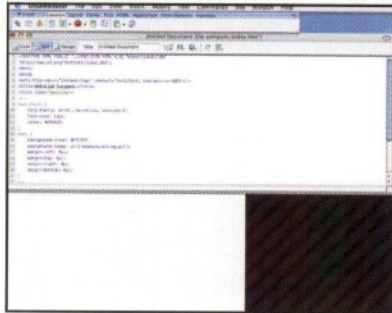


Part 4: Creating the content

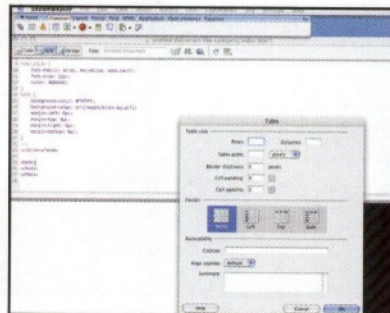
Inserting the main imagery and text into the page...

Dummy text

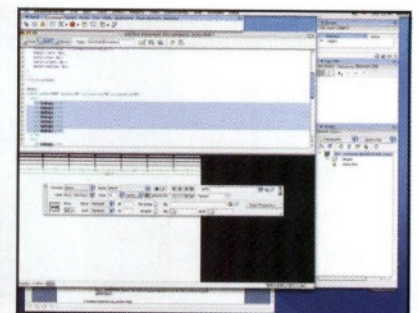
When creating web pages, it's a good idea to insert dummy text (if the copy is not written) to give an idea of how it will look – and to make sure it works! Lorem Ipsum is the dummy text of the printing and typesetting industry, and is easy to copy into your document. There are numerous Lorem Ipsum generators online – try [w] www.lipsum.com, which is an easy-to-use text generator.



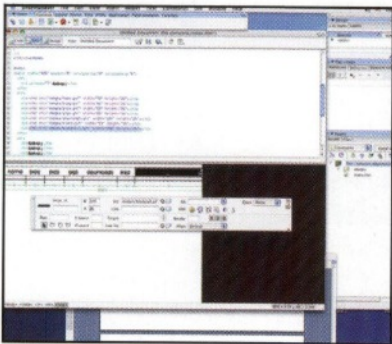
1 There are three options when it comes to viewing your Dreamweaver document: Code view, Code and Design view, and Design view. These views can be changed by clicking on the three different buttons at the top left of the document.



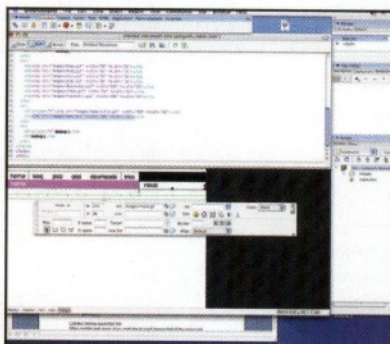
2 Select the Code and Design view. Generic 'page setup' HTML code can be seen in the Code view, and the Design view will be blank. Click on the Design view within the document and select Insert Table from the Common tab within the Insert panel.



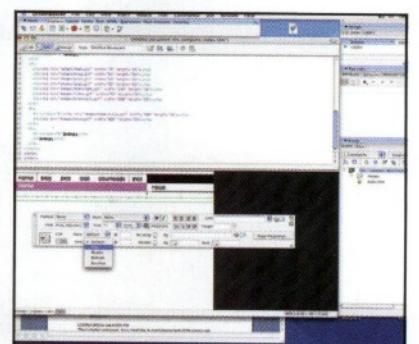
3 Set rows to 4, columns to 7 and width to 600px, and click OK. A table will be inserted into your document. (Click on 'table' in the bottom bar of layout view and note the code added in the code view.) In Design view, select the top row by clicking and dragging the mouse over the cells. Click the Merge button on the Property Inspector.



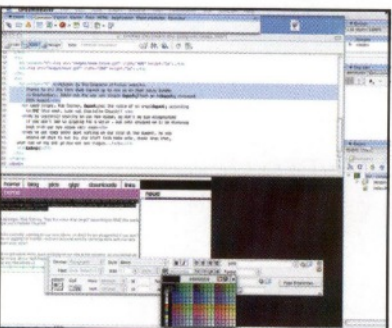
4 Click in the first cell of the second row and click on the insert image icon on the Insert panel. Navigate to home.gif within your images folder and click Insert. Do this for the next five cells in this row, inserting its relevant navigation gif to correlate with your mock-up design. Click in the last cell of this row and insert blackcell.gif from the disc.



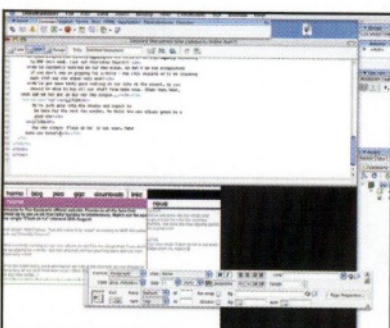
5 Select the first six columns in the third row and click merge. Do this again for the bottom row. This should leave you with two columns in both rows. In the third row, insert home-title.gif in the first cell and news.gif in the second.



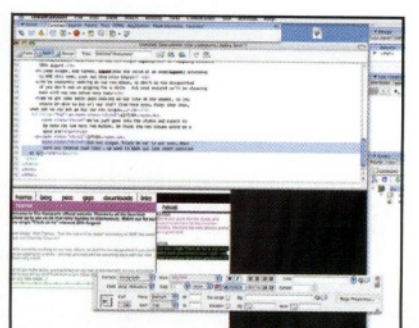
6 Click in the left cell on the bottom row and use the dropdown in the Property Inspector to vertically align the cell to Top. Start typing all the information you want, using [Return] for a new paragraph and [Shift] + [Return] for a new line.



7 Select the first paragraph and in the Property Inspector set the text colour to black (#000000) and bold.



8 Click on the bottom-right cell and set it to vertically align to Top. Start typing all the information you want, using [Return] for a new paragraph and [Shift] + [Return] for a new line.



9 Select the date texts and in the Property Inspector set their text colour to black (#000000). Select the subsequent paragraphs and set their colour to pink (#E725C9).

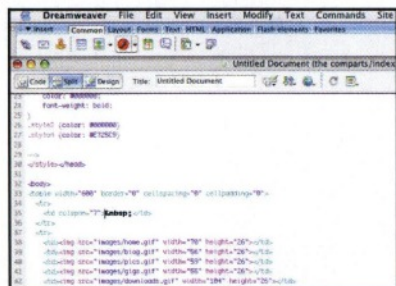


Part 5: Bringing it to life

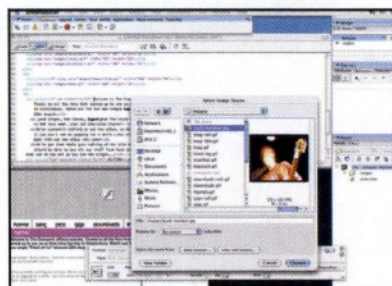
Add the Flash file and position images within the page...

Preview in browser

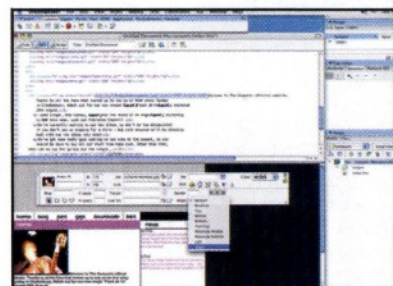
While working on your page, you can check to see how it looks in a browser at any time (Dreamweaver's Design view may not always display it 100-percent accurately). Choose File > Preview in Browser... Then select the browser in which you wish to preview your file. Edit Browser List does exactly as it suggests, and enables you to add more browser options to this list.



1 To insert the Flash swf file, select the top table cell within the Design view. From the Insert bar click the red Flash icon, and navigate to your swf file. Click Choose, and Dreamweaver inserts the Flash file and all the code that it needs to run.



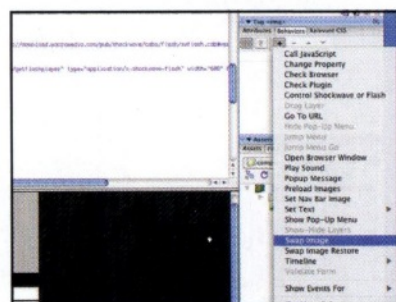
2 Click at the start of the text in the bottom-left cell. Go to the Insert panel and click on the insert image icon. Navigate to band-member.jpg in the images folder and click Choose.



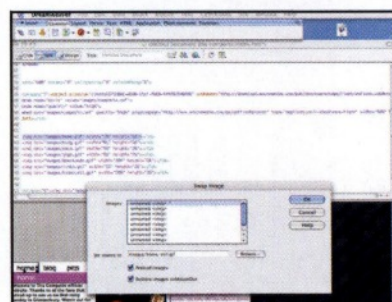
3 Click on the inserted image of the band member and in the Property Inspector select Right under the Align dropdown. Still in the Property Inspector, give the image a Vspace and Hspace of 10px.

Part 6: Simple JavaScript rollovers

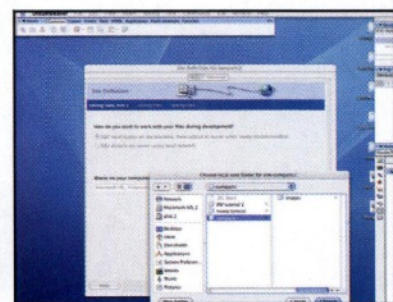
Apply effective rollover states to your navigation and add the finishing touches...



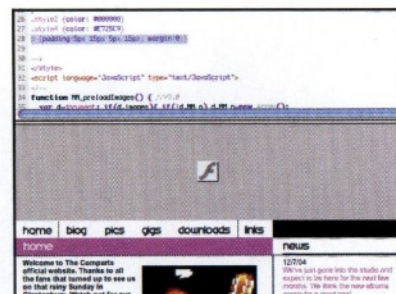
1 Creating the rollover effects for the navigation can be done by using Dreamweaver's preset JavaScript behaviours. Select the 'home' navigation image, and from the Behaviours panel click and hold on the [+] icon. From the resulting list, choose Swap Image.



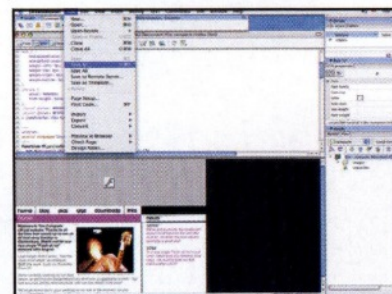
2 From the next window, use the Browse button to locate the image you wish to swap to on rollover. This should be in the images folder within the Compartments folder you created. Make sure 'Preload images' and 'Restore images onMouseOut' are both checked. Click OK.



3 Continue to do this for the rest of the navigation images. You'll notice that Dreamweaver has inserted all the appropriate JavaScript code into your document. This obviously saves you a lot of time, and requires you to know very little about JavaScript coding.



4 To tidy up your paragraph text, you can insert an additional style into the style code that Dreamweaver created in the header. As the paragraphs are contained within 'p' tags, you can set a style for these. We gave this style a margin value of 0, and set the padding to 5px top and bottom, and 15px left and right.



5 The easiest way to create the next pages for your site is to simply choose File > Save As, and save your page as biog.htm. This gives you the starting point for this page, and all you need to do is re-write the text and change appropriate images.



6 To make the navigation images active links, click on one (biog, for example), and from the Property Inspector click on the folder icon next to the Link field, and navigate to the file you wish to link to – in this instance, biog.htm. Click Choose, and Dreamweaver inserts the relevant code. ■



Inspirational new design techniques!

Designing a web site can be a frustrating affair. With the tremendous amount of powerful tools at our disposal, you'd think that a blank canvas was a dream come true for all but the most incapable of web designers. Unfortunately, as with most things in life, it's not as simple as that. What about site functionality? What about the bandwidth of the target audience? Usability? Accessibility? **Tom Knowles** leads you through the minefield and shows you the way to inspired design...







Finding inspiration is a lot like panning for gold; it requires effort, it can take a long time and you've got to know where to look but, once you've found it, the rewards are beautiful. In the same way, over time, you will learn to recognise where inspiration really comes from so, whenever you need it, you will know where to look.

Inspiration from your peers

Your first port of call will logically be the work of the web's most accomplished designers. Eric Jordan and his company 2Advanced have consistently churned out beautifully crafted examples of futuristic, high-impact motion graphics and applications. However, their influence has been such that they are also, unfortunately, one of the most plagiarized agencies in the industry. More on that, later.

Other notable agencies whose work is considered to be amongst the best include Fantasy Interactive (www.fantasy-interactive.com), Flashlevel (www.flashlevel.com), Firstborn Multimedia (www.firstborn-multimedia.com), The WDDG (www.wddg.com), and Joshua Davis Studios (www.joshuadavis.com).

2A's work, and the work of many other great web designers, despite its continual technical innovation and joyous execution, is still, however, grounded by a conventional approach. Acclaimed designer Yugo Nakamura, on the other hand, relentlessly pushes back boundaries, not just in terms of Flash/Shockwave scripting, but in terms of what we expect from the web. He creates stunningly innovative interfaces and environments, some of which are so ahead of their time that the only home for them is on his personal site (www.yugop.com). His concept is that we've been using the web to mimic things that have already been done in other mediums, print, broadcast, etc. He says that we should instead take advantage of the fact that we have the opportunity to create purely for the computer; to build things that could only work within a computational environment.

In fact, the biggest inspiration to be drawn from him,

Your own photographs can be a great way to get the creative juices flowing



Flashlevel might tend to stick to one design formula, but the quality shines through (www.flashlevel.com)

If ever there was a site designed to inspire creative thinking, it's www.yugop.com

beyond his technical achievements, is that large clients such as Sony and NEC routinely hire him to express their ideas in any way he chooses. That's got to be a good thing, because when the big players are willing to take risks, it means that us mere mortals have no reason not to be as experimental as possible.

Inspiration from other sources

If you're in the right frame of mind, inspiration is everywhere. If you look at what makes up a web site (images, typography, user interfaces), you'll realise that those core areas can be found at the base of almost everything around you.

Take typography, for example. It's used to brilliant (or terrible) effect on everything, from crisp packets to film credits, from road signs to newspapers. Take a closer look at the fonts used and their different purposes. You'll start to

TEN DESIGN STYLES TO FEED YOUR IMAGINATION

OVER THE NEXT FEW PAGES, YOU'LL FIND EXAMPLES OF TEN OF THE MOST SUCCESSFUL, MOST RECOGNISABLE OR MOST COMMONLY USED SITE STYLES. FEEL INSPIRED.

1



Futuristic

The sleek, ultra-detailed and technical style, as seen on countless web sites these days.

Pros:

High-impact and visually impressive.
Fairly simple to achieve.
Lends itself well to lots of purposes.

Cons:

Requires lots of attention to detail.
Very commonplace.
Doesn't have a very flexible tone.

www.2advanced.com

2Advanced was an early purveyor of the futuristic style.

2



Pixel Art

Informal and slightly kitsch, pixel art looks like the computer game artwork of yesteryear.

Pros:

Simple and appealing.
A winner with those who've grown up with computers.
Instantly lends a friendly tone.

Cons:

Hugely time consuming.
Graphics aren't scaleable.
The tiny elements it uses can cause usability issues.

www.tongsville.com/city/index.html

Built by Preloaded (www.preloaded.com), Tongsville contains some great examples of intricate pixel art.

see rules appear, and when you do, you'll also begin to easily differentiate between good and bad typography.

User interfaces are also everywhere you look, it's just that we do sometimes take the truly great ones for granted. A book has a user interface. A door has a user interface. A microwave has a user interface. Naturally, some of these interfaces won't have much in common with the average web site, but more often than not you'll be able to draw new ideas about how your users can interact with your site.

Inspiration vs plagiarism

One of the great things about designing for the web is that there's lots of help available. The community is close-knit enough to mean that the big forums (www.ultrashock.com, www.flashkit.com, www.were-here.com) are frequented and contributed to by all the big names, and each new web site is scrutinised and critiqued to the nth degree. This means that you're never far from help, opinions and constructive criticism, no matter whether you're just starting out or a seasoned pro.

However, this intense focus on the industry's work — by the people who produce it — creates an interesting effect. At its most beneficial, it pushes everyone's skills, and the industry as a whole, forward. A big-name designer will come up with a technique so original and so superb that everyone will rush to discover how it was achieved, and naturally attempt to achieve it themselves. At the other end of the scale, though, is the wholesale rip-offs attempted by some of the more unscrupulous designers equipped with Flash file decompilers and a penchant for delving into their browser cache. As we mentioned earlier, 2Advanced is the biggest victim of this simply because its work caused such a huge ripple when it first arrived, and it is now synonymous with drop-dead gorgeous Flash sites.

But when is inspiration just thinly disguised plagiarism? The problem is that this claustrophobic community of ours can tend to become overly hysterical. There seem to be hundreds of members of both Ultrashock and Flashkit whose whole purpose in life — it

OUTSIDE INFLUENCES TEN TOP WAYS TO GET INSPIRATION WHEN YOU'RE OUT OF THE OFFICE

1 Cinema

Films often have memorable visual themes; eg *Bladerunner*, *2001: A Space Odyssey* and *Se7en*

2 Travel

Expose yourself to a variety of sights and sounds.

3 Cities

Cities can be rich sources of imagery, atmosphere and culture.

4 Newspapers and magazines

If you're looking for typographical or photographic inspiration, start here.

5 Books

Books naturally evoke imagery and tone, both of which are crucial elements of design.

6 Nature

Who says that design has to have straight lines or be mechanical, cold or unnatural?

7 Packaging

Packaging is another rich source of eye-catching typography and commercial techniques.

8 Architecture

Both contemporary and classic architecture can inspire ideas about structure and form.

9 Fashion

The fashion industry makes great use of photography in its advertising.

10 Technology

Even everyday technology has a huge amount of design thought behind it.

Something as simple as a photo of the sky can get you thinking

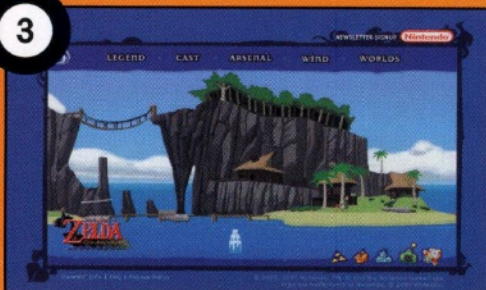
Pen and paper should never be far from your computer. Get scribbling!

seems — is to hunt around for low-key personal sites and accuse their owners of ripping off 2A, or Firstborn, or whoever. But they're missing the point.

True plagiarism, using the work of others and passing it off as your own — harms only the integrity of the plagiarist. The real talents and innovators are going to stay at the top of the tree because they're the ones pushing the boundaries. And besides, the reason you don't see them bleating is because they're too busy working on their next project to be worried about some young kid who's stolen a couple of their GIFs.

On the other hand, walking in the footsteps of others certainly isn't plagiarism. The great thing about the web design community is its commitment to the sharing of knowledge: the huge number of open-source download sites is testament to this. Books and courses can only

3



Cell Shaded/Cartoon

Popular at the moment in the Japanese anime style; cell-shaded graphics can look excellent when deployed well.

Pros:

Informal and youthful.
The vector approach is scaleable to other uses such as print.
Appeals to a wide audience.

Cons:

Requires good illustrations.
Animations can take a long time.
Can seem too informal.

www.zelda.com/gcn

In tandem with the Gamecube title, the Zelda web site uses a cell-shaded design to great effect.

4



Corporate

The clean and business-like style favoured by most large, household-name companies.

Pros:

Easy to put together.
Professional and clear.
Unlikely to offend.

Cons:

Can fail to excite visitors.
Needs a large amount of content to add weight. Requires a strong identity to make up for lack of visual panache.

www.samsung.com

Although rather corporate and conventional, Samsung's site contains lots of more interesting areas, notably some top-notch Flash applications.

5



Cutting edge

Contemporary styles are obviously subject to change, but fresh, colourful and clean sites are currently in vogue.

Pros:

A cutting-edge look and feel can make your site stand out. Appeals to a wide demographic. Likely to be plenty of sites to draw inspiration from.

Cons:

If everyone's doing it, you could end up looking unoriginal. Cutting-edge visuals often require cutting-edge skills and cutting-edge software.

www.hybridworks.jp

This Japanese portfolio site has everything: great typography, beautiful visuals and tightly executed coding.



teach you so much; you have to learn by doing and this often entails learning the techniques of others. If, on occasion, this means that beginners can tend to mimic the established style of more prominent and experienced designers, then so be it. That you don't see this in high-profile, professional sites is precisely because the designers involved are experienced and skilled enough to have developed their own style.

In any sort of design, you cannot be expected to find your own voice before you've studied the styles and techniques that have gone before. Why should web design be any different?

What does the style say?

One of the common mistakes people make is to force a style (usually the most popular one in the industry at the time) on the project they've been given. You'd think it would be ridiculous to design a flower shop or accountants' site in the futuristic 2Advanced style, but it really does happen, especially amongst inexperienced designers who are keen to show off their skills.

If your client is an architect renowned for their ultra-minimalism, it's no good wandering down to a local anime festival in order to brush up on your illustration techniques. The crucial point to remember is that you must, above all else, base your design decisions on the client's objectives and the site's overall purpose. Only then can you go looking for further inspiration.

Turning inspiration into reality

You're inspired, but how do you transform your inspiration into a working project? One of the most important aspects of web design is that you must always remember who and what you're designing for. In their quest for better inspiration, designers often forget this and come up with some wild and wonderful, high-concept ideas in the initial stages of a project, never really stopping to consider the web site's target-audience or the practical purpose of the site.

Sometimes it's a case of putting your inspiration into

BINARY FINERY TOP TEN PLACES FOR ONLINE INSPIRATION

- 1 Ultrashock**
The place to go for discussion.
www.ultrashock.com
- 2 Favourite Web Site Awards**
The very best web sites out there.
www.favouriteweb.siteawards.com
- 3 Flashkit**
A great place for beginners.
www.flashkit.com
- 4 Designers Republic**
Cutting-edge design discussion and links to hundreds of portfolios.
www.thedesignersrepublic.com
- 5 Pixel Surgeon**
A digest of the creative and interesting – including web sites, films and music.
www.pixelurgeon.com
- 6 K10K**
A global design portal, updated hourly.
www.k10k.net
- 7 Linkdup**
Great site links.
www.linkdup.com
- 8 Getty Images**
Pricy, but Getty is the web's best and most searchable photography resource.
www.gettyimages.com
- 9 Chapter 3**
Jens Karlsson's personal site showcases his stunning 3D digital artwork.
www.chapter3.net
- 10 Gmunk**
Jaw-dropping video work from motion graphics artist Bradley Grosh.
www.gmunk.com

Nature is often overlooked as an angle into creative work

perspective. You may have been inspired by a recent trip to a local forest, and have subsequently had some fairly crazy ideas about transporting your users to some fully interactive, digital woodland. It's at this point that you have to be your own editor and reign in some of your wilder ideas.

An interactive forest is probably going to be pretty impractical, even if you did have a massive team of people on hand to build it to the precise standard you can envision in your head. Instead, focus on how the tone and visual style of your inspiration could affect a more traditional site layout. Why not keep a fairly standard structure, but use branches of a tree as the navigation bar? How about using Flash to create a header bar containing swaying trees and falling, autumnal leaves? No matter what the style, no matter what the content, a web site has to behave in a usable manner. At the end of the day it has to be used by its visitors rather than to just annoy them and drive them away.

Some styles of web site are going to require a lot more work than others, and some are inevitably going to need more specialised software. This is clearly going to have an impact on the style you choose to work in. If you're

6



Organic

The richly graphical style pioneered by the likes of 24-7 Media.

Pros:

Distinctive and visually impressive. Ideal for strong identities. Works well with a broad range of typography and colour.

Cons:

Requires expert use of Photoshop or Fireworks. Not particularly subtle. Usually means some increased download times.

www.24-7media.de

24-7 Media specialise in lush, detailed interfaces and its home page is no exception.

7



Retro

Often applied in a humorous context, retro styling can provide a distinctive voice for any site.

Pros:

Light-hearted. Ideally suited to humorous content. Works equally well with or without animation.

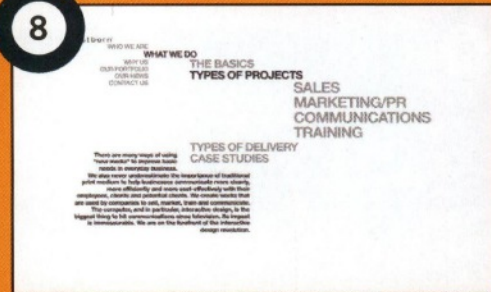
Cons:

Difficult to make authentic. Too informal for serious content. Could go over the heads of some younger users.

www.altoids.com

Altoids' site (built by the WDDG – www.wddg.com) is a hugely enjoyable affair, based around 1950s-style visuals.

8



Typographic

Typographical sites usually dispense with graphical embellishments and instead focus on using typeface choice and composition to do the talking.

Pros:

Ultra-clean and stylish. Clear and concise. Usually produces small file sizes.

Cons:

Can seem boring to some users. Requires good use of language. Decent, distinctive fonts can be fairly expensive.

www.firstbornmultimedia.com/flashLarge.htm

Combining superb ActionScripting with clean, elegant typography, FirstBorn Multimedia's site is a real source of inspiration.



Some of the work out there is so mind-blowing, you cannot fail to be inspired. www.gmunk.com is a case in point



Most designers listen to music while they work. Maybe it's time you got some new tunes on your iPod (www.apple.com/itunes)



Pixelsurgeon: a digest for designers – sites, film and music

planning on doing an illustrative, cell-shaded site, you're really going to need a pretty decent illustration package such as Adobe Illustrator or Macromedia Freehand. Likewise, if you want to create some rich, layered montages using photos, shapes and text, you're going to need Fireworks or Photoshop.

The key to all this is that unfortunately, certain styles are quite elitist: with some designs, you really can't hope to achieve much without the right tools. This isn't always a bad thing, however; it can force you to focus more on the basic foundations of your design: the colours, fonts and geography of the layout. As a beginner, focusing on these important and fundamental principles rather than playing around with advanced tools can be enormously beneficial. There's nothing worse than a web site that has been created by someone with no eye for design whatsoever, who has just gone nuts with a copy of Photoshop and Swift 3D.

So you've finished your latest project. You had a blank canvas and a blank mind to begin with, but you found

inspiration and you've now got a finished project in front of you. Okay, so you're entitled to feel pretty smug for a while. But then it's time to start asking yourself some blunt questions: Does it look and perform how I envisaged it would when I originally found my inspiration? Or did my original inspiration end up having a negative effect on the site? In fact, can you even detect my original inspiration in the design? Or has it got lost somewhere? Have I paid enough attention to details? Is it actually any good?

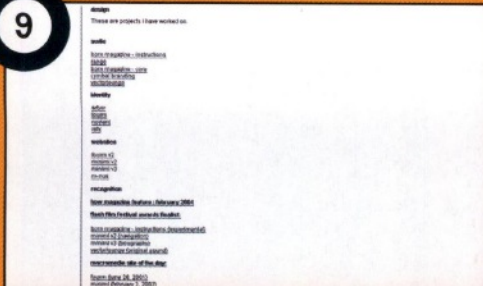
This is not just a visual thing, it can be about functionality, too. If you've spent time trying to build a similar Flash effect to one you've seen elsewhere, revisit the original version and compare the two. Does your version come up to scratch, or better yet, has it improved on the original? Or is it a poor relation?

Every designer must go through this process of self-evaluation and criticism. There's no use in resting on your laurels; you have to constantly re-evaluate your skills – after every project – if you want to keep improving and become a great web designer.



Magazines and newspapers often carry superb examples of typographic and photographic work

9



Minimalist

The less-is-more rationale favoured by many designers, minimalism can work wonders when content is king.

Pros:

Quick to produce.
Direct and unfussy.
Ideal for low-bandwidth.

Cons:

Can look sparse and sometimes lacking in sparkle.
Can be difficult to use – needs a design-savvy audience.

www.miniml.com

Font designer, Craig Kroeger's personal site focuses purely on his excellent array of pixel fonts.

10



Handdrawn/Scanned

A more traditional, back-to-basics approach, whereby hand-drawn elements are integrated into the interface.

Pros:

Warm and engaging.
Completely original.
Lots of potential for humour and personality.

Cons:

Drawing and scanning can be incredibly time-consuming.
Requires a strong idea.
Has a tendency to be self-indulgent.

www.nanahiro.com/compe

Beautifully made and nothing short of bonkers, this site takes you on a bizarre puzzle-solving journey.

Act on your inspiration: turn over for a tutorial on how to turn your ideas into a web site





ACT ON YOUR INSPIRATION

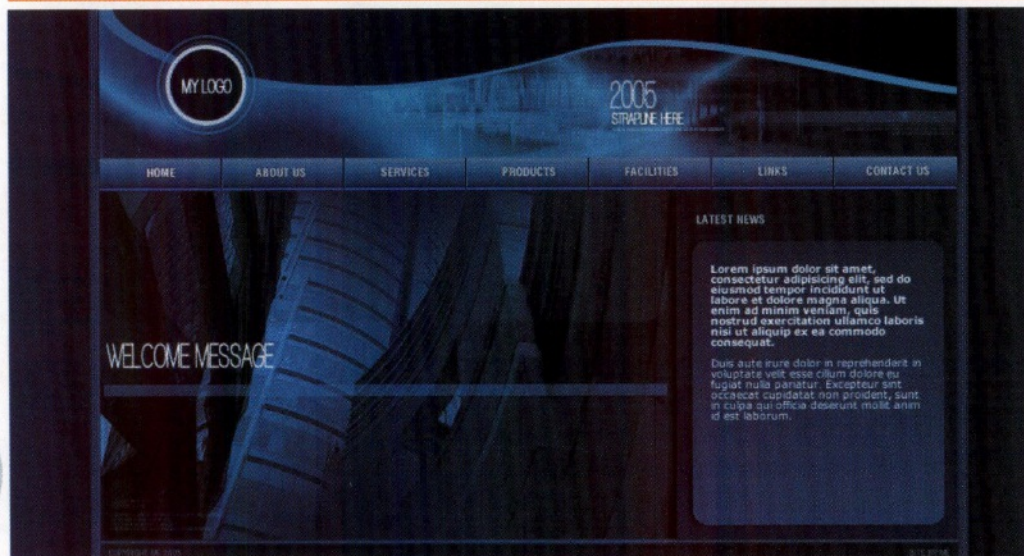
MAKING AN INITIAL IDEA INTO A WEB SITE

So you have taken some time to think about what you want to achieve with your new site. You have decided on your theme, or your client has handed over the brief, and you are sure that you have the appropriate tools to adequately render your vision.

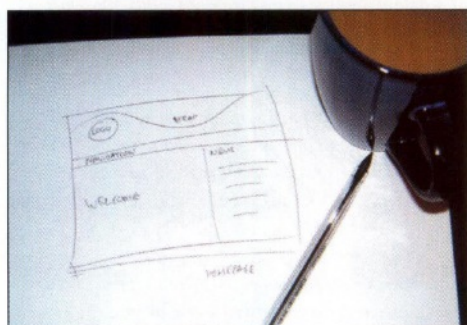
What next? Tom Knowles demonstrates the process of bringing a pen and ink idea to the liquid crystal screen.



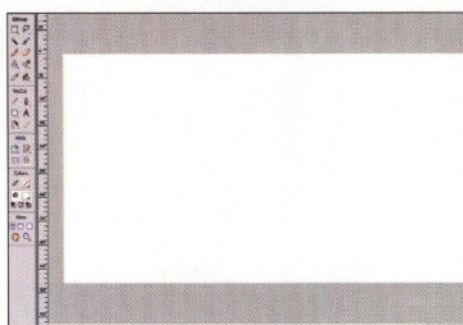
THE FINAL DESIGN



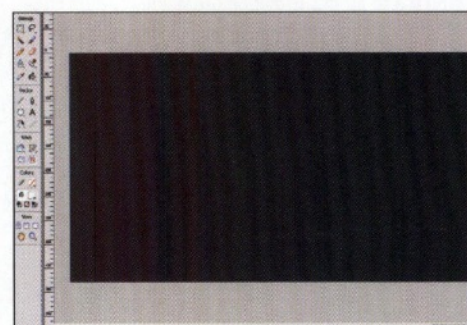
TUTORIAL – THE SITE CONSTRUCTION PROCESS



1 Before we even switch our computer on, we need to pick up a pen and paper and sketch the basic web site layout.



2 Once we've got a clear idea of our interface's geography, we can create a blank canvas in Fireworks and start designing.



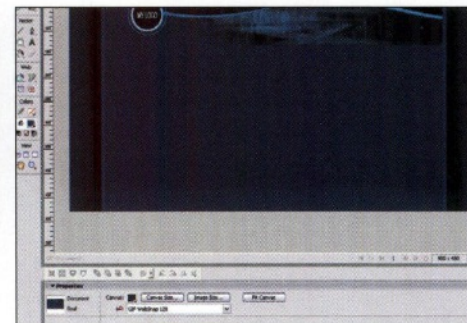
3 We're going to design a futuristic site; but first we'll need to come up with a colour scheme; which in this case is blue.



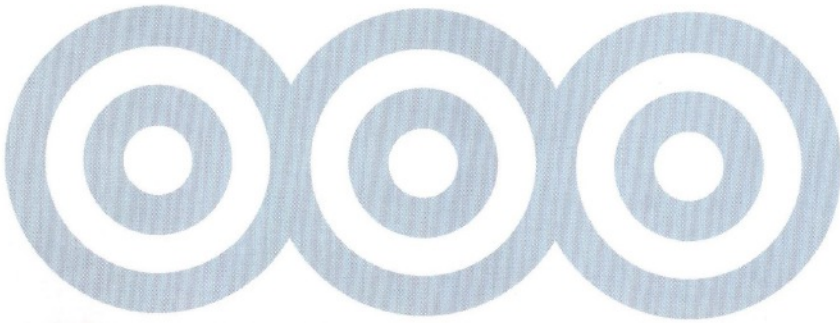
4 Using some basic rock textures (which are freely available all over the web), we can create some subtle site borders.



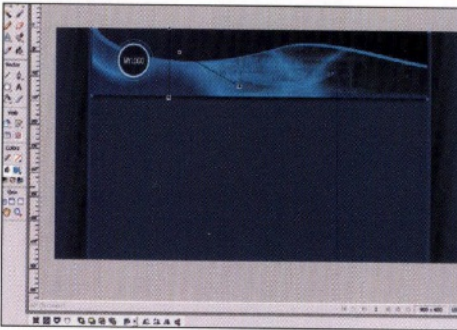
5 Fireworks' excellent fill textures allow us to add subtle embellishments. By this point we should also be starting to build up a colour palette.



6 Gradient-filled swooshes are ideal for creating layers in a web site's header graphic, especially when used in conjunction with cityscape images.



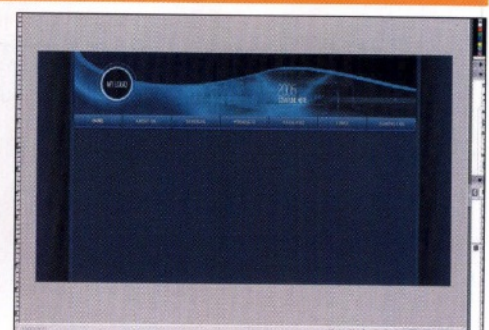
TUTORIAL – THE SITE CONSTRUCTION PROCESS continued



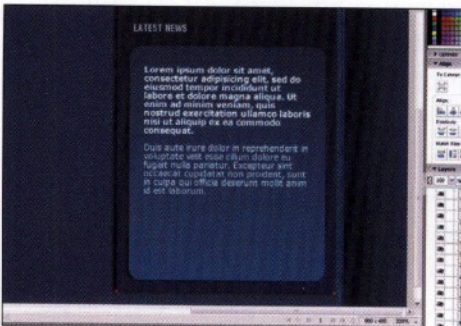
- 7** Fireworks offers a rich variety of gradient fills. When used subtly, these can help you to build up sophisticated textures.



- 8** Futuristic styling often incorporates very small patches of text. This sort of attention to detail enhances the technical tone.



- 9** When designing navigation bars, one of the most important things is to ensure they are uniform and, therefore, easily changeable.



- 10** These days, most sites' home pages worth their salt have some sort of news feed on them. Our example is no exception.



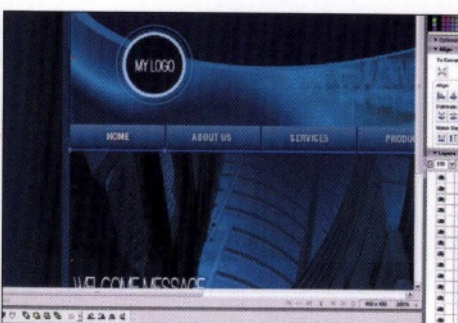
- 11** Imagery on this type of site is extremely important, and usually falls into one of two categories: architectural or abstract 3D.



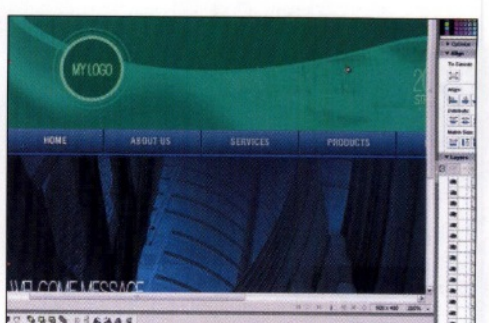
- 12** Again, using Fireworks's gradient fills and textures, as well as details such as transparent vector shapes, we can build up some rich graphics.



- 13** Once you've created your overall layout, it's always a good idea to tweak details (fonts, colours etc) here and there.



- 14** The next stage involves analysing the design and working out how to go about transforming it into HTML and/or Flash elements.



- 15** Fireworks provides plenty of functionality to help cut up the design into more manageable chunks, including slicing and compression tools.



Designing for Broadband



If you were to compile a list of all the words people chuck around when talking about the internet these days, broadband would be right up there with iTunes, blogging and spam. It seems as though the idea of dial-up is becoming increasingly obsolete and, as the web increases its stranglehold on everyday life, everyone is rushing to get the fastest, fattest connection possible.

Broadband connection figures are rising by the day, and the Government is insistent on getting coverage up to 100 per cent within the next couple of years. As prices continue to come down and speeds go up, connections that historically you would only experience if you worked in a big company are now cropping up in the nation's households.

What this means, of course, is that we're well on the way towards the next phase of the internet's development, or Internet 2.0 as some trendy cyberfolk like to call it. Dial-up connections and all their problems are fast becoming the exception rather than the norm, and this in turn is making yesterday's cutting-edge technology available to everyone. **Tom Knowles** examines the issues currently facing broadband site designers...



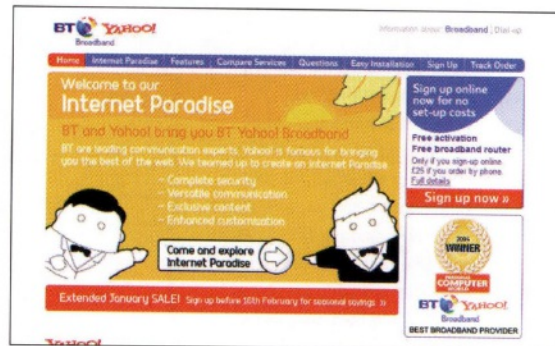
All the components needed to complete this tutorial can be found on the cover disc.



The changes in internet technology in the home indicate changes for web designers and developers as the broadband audience ceases to be such a niche market. Indeed, with increasingly huge numbers of people around the world using the web as a part of their job, as well as faster connections at home, the average user is now fully broadband-enabled. So it's time for a reappraisal of what the average person wants and expects from their web experience. It's not just connection speed that has changed; Joe Average is now more technologically-savvy, more experienced and in most cases more demanding.

One of the problems facing us is that whilst broadband is more widespread than ever, the phrase itself is rather woolly. So woolly, in fact, that it's become almost a relative term; if you're used to a fast connection at home, you're probably going to consider the office connection to be broadband. Whereas, if you've been used to a cranky old modem, even a lowly 256kbps connection will seem like lightning. The official word from Ofcom is that speeds upwards of 128kbps can be called 'broadband'.

There is of course a huge amount of discussion about what broadband packages are available and at what price: wherever you go, common small-talk discusses who's got the most bandwidth. Availability and price are changing rapidly, but the current crop of home packages from the big providers start with 256kbps connections (roughly around five times the speed of the average modem), and go up to 1Mbps. There's also a raft of options related to usage allowance, something which can greatly affect price. Not to mention the fact that due to connection sharing (everyone on your street using the same stretch of cable, for example), you're unlikely to achieve maximum speed.



BT Yahoo! is one of a number of ISPs providing a variety of broadband speeds

Technology

Aside from anything else, a broadband audience means that certain technologies and techniques are no longer off-limits. In its early days, streaming audio used to be a choppy, frustrating affair because most of us were using it via a modem connection. These days, it's child's play to view even the slowest broadband connections. The BBC, whose *raison d'être* is accessibility, has embraced the concept and is now streaming all its radio content from the BBC site. As with most BBC projects, it's extremely well executed and, in addition, it's also a great idea that wouldn't be possible without the prevalence of fast connections.

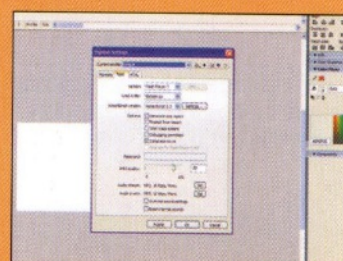
As developers, projects such as the BBC's radio centre require us to make decisions about how we deliver our solutions. In the days when broadband was really only the preserve of tech-savvy ➤

TUTORIAL USING FLASH VIDEO

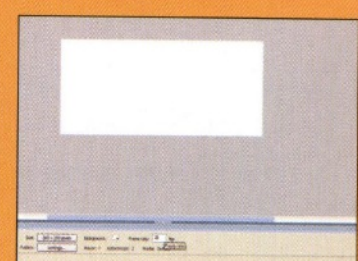
DISCOVER HOW TO CREATE YOUR OWN FLASH VIDEO USING FLASH MX 2004



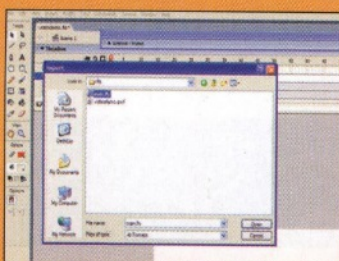
1 Flash MX 2004 introduced the concept of Flash video. In this walkthrough we'll look at how to import and control it.



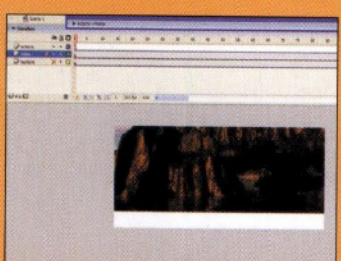
2 Open Flash, and choose New Flash Document. Now choose File ➤ Publish Settings ➤ Flash, and select ActionScript 2.0.



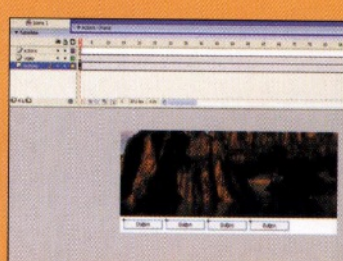
3 In the document properties panel, set the dimensions to 560x250 pixels, and the frame rate to 30 frames per second.



4 Create a new layer on the timeline and choose File ➤ Import ➤ Import to Stage, and browse to the main.flv file (included on the CD).



5 The video file will appear on the stage, and spread over 523 frames. Create two new layers with the same number of frames.



6 Call one of the layers buttons, and drag four instances of the button component from the components window onto the stage.



7 Give the buttons the instance names startbutton, stopbutton, playbutton and endbutton. Return to the first layer and open the actions window.

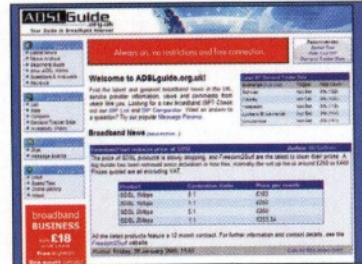


users, you could rely on them to handle whatever content you threw at them.

Back then, media players such as RealPlayer and Quicktime competed on a level playing field for adoption. Of course, since streaming content has become more widespread and more accessible, it has attracted giants like Microsoft with its Windows Media Player and file formats. What this has done is crowd out players like Real. Microsoft (and Apple with Quicktime) can and do bundle their players with their operating systems for free, meaning that smaller, commercial players such as Real simply cannot compete.

This does make things easy for the developer though – regardless of what you may think of the actual technology, you can pretty much guarantee that your users will have either the Quicktime player or Windows Media Player installed, and hence your choice of delivery methods is narrowed.

On a side note, Macromedia has been quietly plotting to muscle in on Microsoft and Apple's duopoly. By building up the video functionality of Flash, they've effectively created a single, flexible media player, of which 97 per cent of web users have, at least, a base version. Unless someone else



ADSLguide.org has up-to-date information on what services are out there and what they cost

develops a worthy alternative, it's possible that in the next year or so the Flash player will become the standard plug-in for all web-based content.

The Flash 7 player itself has revolutionised attitudes. Take Ford's new vehicles site (www.fordvehicles.com) as an example. It is a mammoth, almost 100% Flash extravaganza. A web site of this size and complexity simply would not have been built were it not for the ubiquity of broadband connections. When an enormous company such as Ford decides that its target market is technologically

ready for such a web site, it's a clear indication that the web is moving forward.

With Flash's increasing power, a lot of people predicted the demise of the Shockwave plug in and its authoring environment, Director. Cleverly, however, Macromedia has slowly developed Director into a heavyweight 3D gaming platform – perfectly in time to respond to the swell in demand for online gaming brought about by broadband.

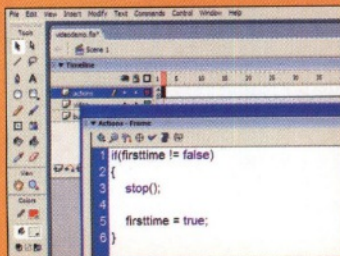
One of the problems we're faced with when developing for broadband is how exclusive or inclusive we're going to be. Swedish agency, Fantasy Interactive, develop solely for Internet 2.0 – their sites are broadband-only.



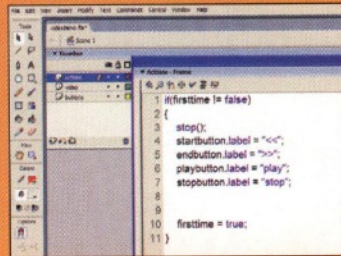
Blueyonder dedicates a specific area of its site to providing broadband content

Unless someone else develops a worthy alternative, it's possible that in the next year the Flash player will become the standard

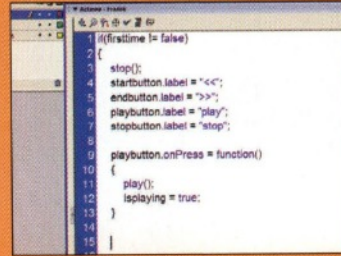
TUTORIAL USING FLASH VIDEO continued...



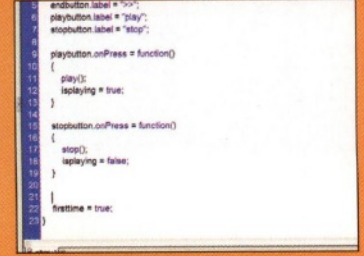
8 Type out the above code. By checking the value of a firsttime variable, we ensure the code only ever executes once.



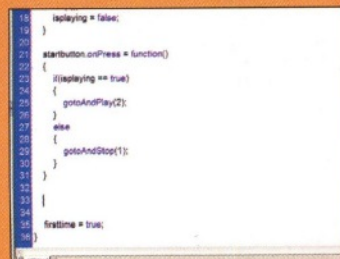
9 The above code assigns labels to the buttons we've placed on the stage. You can also do this via the properties panel.



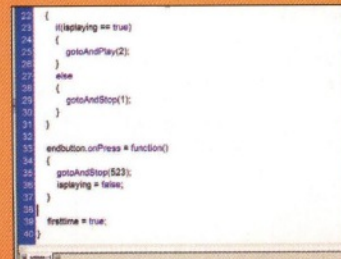
10 This code simply starts playing the movie when the user presses the playbutton. It also creates a boolean variable called isplaying.



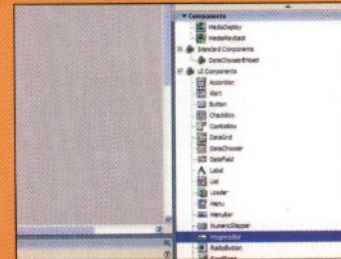
11 The stopbutton stops the movie's playhead, using the stop() action. Notice how we give the isplaying variable the value false.



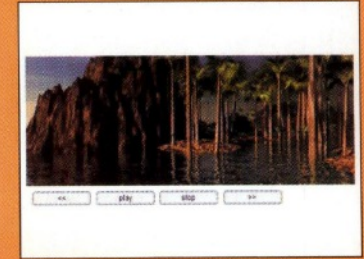
12 The startbutton sends the playhead back to the first frame, and then either starts playing again or stops, depending on the value of isplaying.



13 The last button, endbutton, sends the playhead to the last frame (in this case 523), and stops on that frame.



14 Video tends to be fairly large in file size, so it might be a good idea to either add a preloader or use one of the video components.



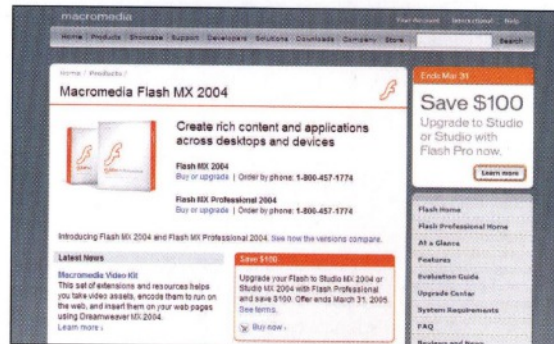
15 Finally, save your file. That's all there is to it. Test your movie by choosing File → Publish Preview → Flash.

and usually require cutting-edge computers and hi-res monitors. This approach obviously works for them and their clients but, in the majority of cases, connectivity and hardware capability don't necessarily go hand in hand. For every net-head with a massive, custom-built machine, there'll be someone using a five year-old corporate PC from their office. Both users may have enormous bandwidth at their disposal, but very different requirements in terms of what they are willing and able to surf.

Trade-offs

Okay, so you've done your research, you know your target audience is young, technically adept and on a decent connection. You're going to put together a top-notch Flash site and knock their socks off. Or so you think. The problem is, however, it's just not as easy as that.

Yes, your user is going to have a broadband connection. As we mentioned earlier, however, that could be a barely-broadband connection that the user is sharing with ten others, or it could be a fat, dedicated T1 line. The point is that despite knowing something about your target audience, you can't assume anything. Just because they might have a faster connection doesn't mean that you can forget about their needs as web users. They still won't want to wait ages for content to download and they'll certainly be



Macromedia's Flash platform. More powerful and more flexible with each release

experienced enough not to want to wade through a bloated and pointless broadband intro movie.

You won't need to spend an age optimising each element to ridiculous levels; the broadband target audience means that you don't have to sacrifice decent quality for the sake of a few extra kilobytes. Time saved in this area isn't time you can afford to spend slurping coffee with your feet up, though. Broadband users expect to get something in return for the fact that you're making



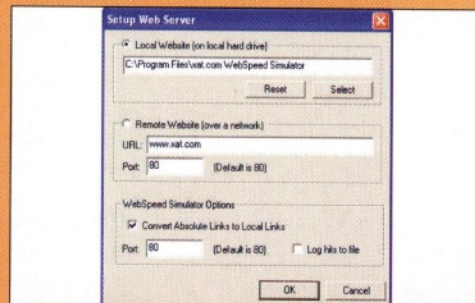
Real's video/audio player has been marginalised by Windows Media Player and Quicktime

TEST YOUR SITE'S BANDWIDTH

WANT TO TEST HOW LONG YOUR SITE TAKES TO DOWNLOAD AT DIFFERENT CONNECTION SPEEDS? TRY WEBSPEED SIMULATOR

When you build a web site, it's vital to know whether it'll load quickly or whether your target audience will grow old waiting in front of their computers to witness your genius. This is especially true for more complex sites using Flash. WebSpeed Simulator (www.xat.com) does exactly as its name suggests: it creates a local bandwidth-limited HTTP server and enables you to watch your site loading in real time with your browser set at different connection speeds. You can see the order in which your site files are being loaded and how long they take. The program is incredibly easy to use and you don't need to change your browser settings or your physical connection to test your site at different speeds. However, as the tool works as an HTTP server, it's not possible to simulate secure servers.

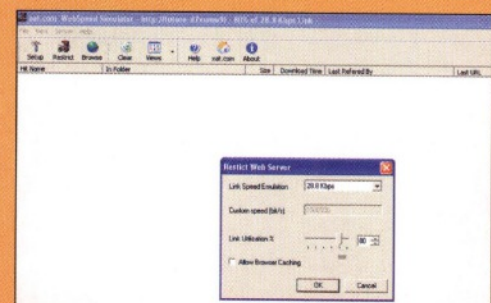
WebSpeed Simulator costs \$99 (around £53) but if you would rather give it a try before you buy, a free 30-day trial is available on this month's CD.



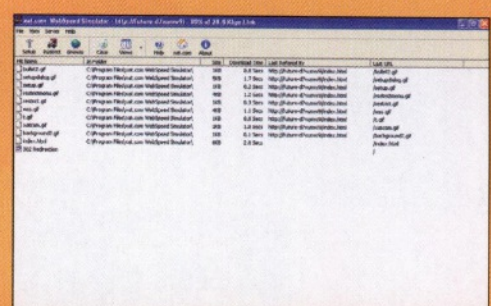
1 If you want to test a site from your local hard drive or a remote site, go to the simulator's toolbar and click the Setup button. Now specify the location of the site and consider clearing your browser cache (in Internet Explorer go to Tools → Internet Options → General → Delete Temporary Internet Files).



3 Finally, click the Browse button. Your default browser will open the site you specified earlier in the set-up process and limit bandwidth to the speed you've entered in the Restrict Web Server dialog. You can now go back to WebSpeed Simulator and adjust the speed to test out different bandwidths.



2 Now click the Restrict button and alter the settings for your simulated connection in the Restrict Web Server dialog. You can change the speed and specify whether you'd like to allow browser caching. If you do this, content already in the browser cache will load as soon as you load a site you've visited previously.



4 WebSpeed Simulator's main window will list the different items and components of your site and display the time each file takes to load as well as the order of loading and file sizes. This way you can easily monitor which parts of your site you could speed up. It's particularly helpful if you work with sites that mix HTML and Flash.



BROADBAND RESPONSIBILITY

ENSURING YOUR SITE IS ACCESSIBLE AND STANDARDS COMPLIANT

On the surface, it might seem that broadband content and accessibility and standards compliance don't mix. This is a dangerous assumption; broadband's increasing prevalence indicates that its content is no longer a niche privilege. The rules of accessibility and usability are more applicable than ever.

The only problem is that since we're still in the infancy of the broadband age, adhering to those rules means that we have to develop some new techniques.

That said, there are still some reasonably obvious concepts that many broadband developers still ignore. Any site, broadband or not, should separate its content from its presentation. By doing this we can give ourselves the ability to provide our content in different ways. This is especially applicable to broadband content: if your textual information is stored in a database, for example, you can easily provide a lightweight text-only version to sit alongside your all-singing Flash site.

Don't ignore usability either, we're in 2005 now and broadband is no longer the sole prevail of designers and IT geeks. Your granny could be using your Flash site and have no idea what to do if she has to navigate the site by chasing an After Effects-generated fireball around the screen.

Take into account your target audience but go ahead and create challenging, exploratory interfaces if it makes sense to do so. The bottom line is, though, that you can't assume broadband users are experts, nor can you assume that they'll get a kick out of your kitchen-sink approach to graphics and sound.

Choosing to develop a broadband site doesn't mean an easy ride. If you want to build something entirely in Flash when it may well suit HTML better, then full marks for ambition, but you're going to have to work hard to close the usability gap. Warner Bros' Road Runner portal is a full-Flash experience, but it still allows you to resize and change the contrast of text.

You might think that certain types of content exempt you from accessibility – think again. You might be delivering streaming Flash video, so take advantage of the fact you can provide optional subtitles. Yes, this usually involves more work, but it's a more responsible and professional approach.

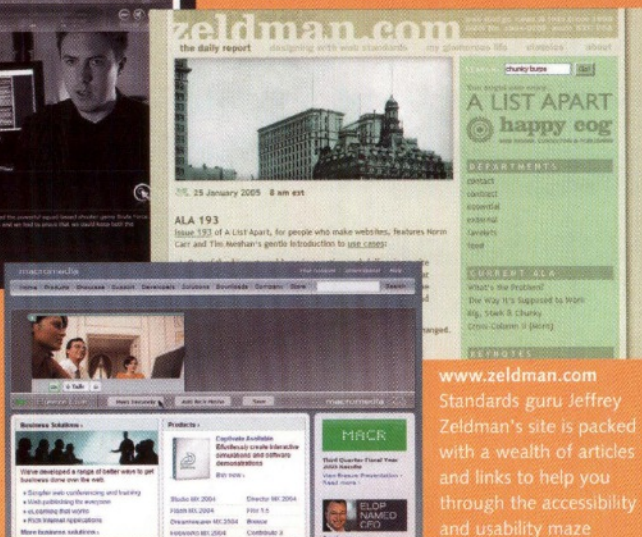
A great example of a really usable and effective broadband site is Fantasy Interactive. The whole site is built in Flash, and the bulk of the content is big, widescreen video. It sounds like a recipe for a usability nightmare, but FI have clearly put a huge amount of thought into making it as usable as possible. They've made sure the site uses the back and forward buttons (rare in Flash sites), as well as providing a customised right-click menu with functionality you'd normally only get with regular web pages (bookmarking etc).

Flash and other broadband-aimed platforms have still got a long way to go in terms of being as browser-integrated as good ol' HTML, but with folks like FI striving to push boundaries, such integration is hopefully just around the corner.

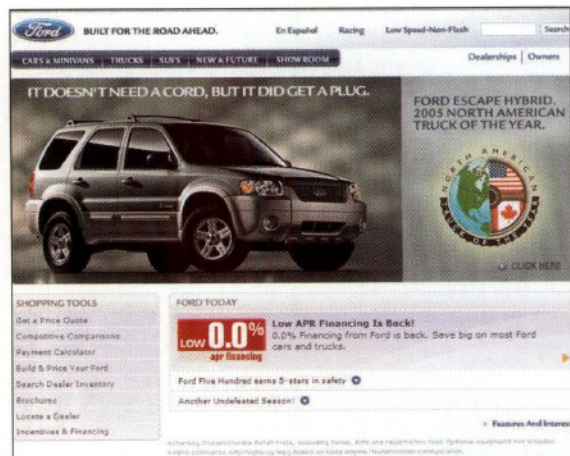
www.fantasy-interactive.com FI's lush new web site is targeted squarely at broadband users, but they've also built it around sound usability principles



www.macromedia.com
Despite being packed with sumptuous multimedia content and rich graphics, Macromedia has done a brilliant job of making its site 100% compliant



www.zeldman.com
Standards guru Jeffrey Zeldman's site is packed with a wealth of articles and links to help you through the accessibility and usability maze



Ford's, cutting edge, full-flash site is an indication of things to come

them wait a second or two longer than they would with a basic HTML site.

As designers and developers, we like making cool stuff. The average modern web user isn't interested in just cool, though. They want their surfing experience to be intuitive, fast and rewarding. If we want them to embrace our cool stuff, we've got to work to make it accessible and – above all – better and more useful than what they've been used to.

Doing so requires work from the industry as a whole: in the same way that the basics of HTML were pushed to create the web as we see it now, the industry now needs to push the current crop of technologies towards mainstream acceptance, and the only way to do that is to make things easy for users.

Cost issues

Currently, it's unlikely that developing broadband-style content for your own personal site or for any fairly low-traffic sites will have a great number of cost implications.

That said, the big drain on server bandwidth is persistent connections, the sort required by streaming and chat servers. So, if you're interested in doing something in that area, you may need to look carefully at what the terms of your current hosting set-up say. Hosting companies such as MediaTemple specialise in multimedia content and, therefore, offer a number of different options for bandwidth usage.

Naturally, the cost implications of designing broadband content are not just associated with hosting. Cutting-edge content requires cutting-edge software, which can get expensive. If



Some companies specialise in hosting large multimedia sites



CLICK FOR MORE TEN GREAT EXAMPLES OF BROADBAND CONTENT

1 Quicktime
www.quicktime.com
The film buff's favourite, Apple's Quicktime site features all the latest movie trailers, streamable via the Quicktime plug-in.

2 BBC news
www.bbc.co.uk/news
The BBC news site might be ultra-lightweight and accessible, but it caters for the broadband audience brilliantly with its live news player.

3 Xbox
www.xbox.com
Microsoft's graphics-laden Xbox site is stuffed to the gills with broadband content, including frequent mini-sites for new games.

4 Roadrunner
www.rr.com/flash/index.cfm
Road Runner is a news and entertainment portal, built in Flash. A staggering technical achievement, the site integrates video and Google searching.

5 Metroid
www.metroid.com
A richly interactive site for Nintendo's Metroid, it delivers a taste of the game via its innovative adventure narrative.

6 Billy Bussey
www.billybussey.com
Billy Bussey is an interactive designer from the US, and his site – despite being incomplete – is a stunning motion graphics showcase.

7 Shockwave
www.shockwave.com
Shockwave has been around for absolutely ages. These days its wide and varied range of games and films is second to none.

8 Yahoo! Launch
uk.launch.yahoo.com
Yahoo!'s long-running Launch site streams a wide variety of internet radio stations in addition to the latest music videos.

9 Gmunk
www.gmunk.com
Gmunk.com is motion designer, Bradley Grosh's personal portfolio, and it features some jaw-dropping After Effects work, streamed via the Quicktime player.

10 BBC Broadband
www.bbc.co.uk/broadband
Winner of the Best Site for Broadband accolade in the 2004 Future UK Internet awards, the BBC broadband site is the perfect combination of news, entertainment, sports, culture and kids content.

you're planning on creating Quicktime video, you'll need a decent video editing package such as Adobe Premiere (around £500) or Quicktime itself (around £700).

Macromedia's Studio MX suite (containing Flash, Dreamweaver, Fireworks and Freehand) is good value, but still a hefty amount of cash (£800). Should you want to develop true 3D visuals, be prepared to pay a premium for the top packages such as 3D Studio and Maya.

Anyone who's familiar with the last couple of incarnations of Microsoft Office will, no doubt, have noticed how many of its features have become web-integrated, and clearly developed with an always-on connection in mind. On the other hand, technologies such as Flash are becoming more and more like desktop applications.

In actual fact, what we're seeing is a convergence of the traditional web and desktop applications, and a new type of multimedia-interfaced, online application emerge. Longhorn, the new version of Windows, is said to be the next step, marrying Flash-style visual behaviour with web application logic. In fact, some observers are saying that the interface system is so powerful that it will overtake Flash as the number one multimedia or application interface environment. Don't bet against it.

The broadband age is exciting – over the next couple of years we're going to see some dramatic developments in the way the web works. With connections becoming faster and increasingly wire-less, we're going to see innovations that we couldn't possibly predict right now. Smarter, all-encompassing devices are going to make us think about what 'broadband content' actually means; indeed, if it means anything at all.

Let's not get ahead of ourselves, however. Fast connections are here to stay, and if you want to stay at the forefront of web design and development, keep in mind this basic rule: a good web site is simply one that does the job it was designed to do really well. ■

THE PROS AND CONS OF BROADBAND CONTENT

THE PROS

1 High-impact

Quality graphics, video and sound can all be used to create high-impact, online experiences, whether these are delivered via browsers, media viewers or desktop applications.

2 No compromises

Since file-size isn't as much of an issue as with narrowband, you don't have to dilute the power of the message you're attempting to communicate.

3 No need for excess optimisation

Broadband allows you to spend most of your time on creating the actual content, rather than optimising it to the nth degree.

4 Forget about being on the web

Some of the best broadband content can make you forget you're on the web – yet another indication that Internet 2.0 is just around the corner.

5 The sky's the limit

Broadband opens a whole new world for developers to investigate, meaning that in many cases the old rules of what the web can do don't apply.

THE CONS

1 Bloated file-size

The target audience isn't connected via paper a cup 'n' string, but that can create the temptation to ignore optimisation altogether.

2 Lack of focus

Good content, ease of use, good ideas. All these things make good web sites, but all too often broadband sites seem to forget about simplicity.

3 Bells and whistles

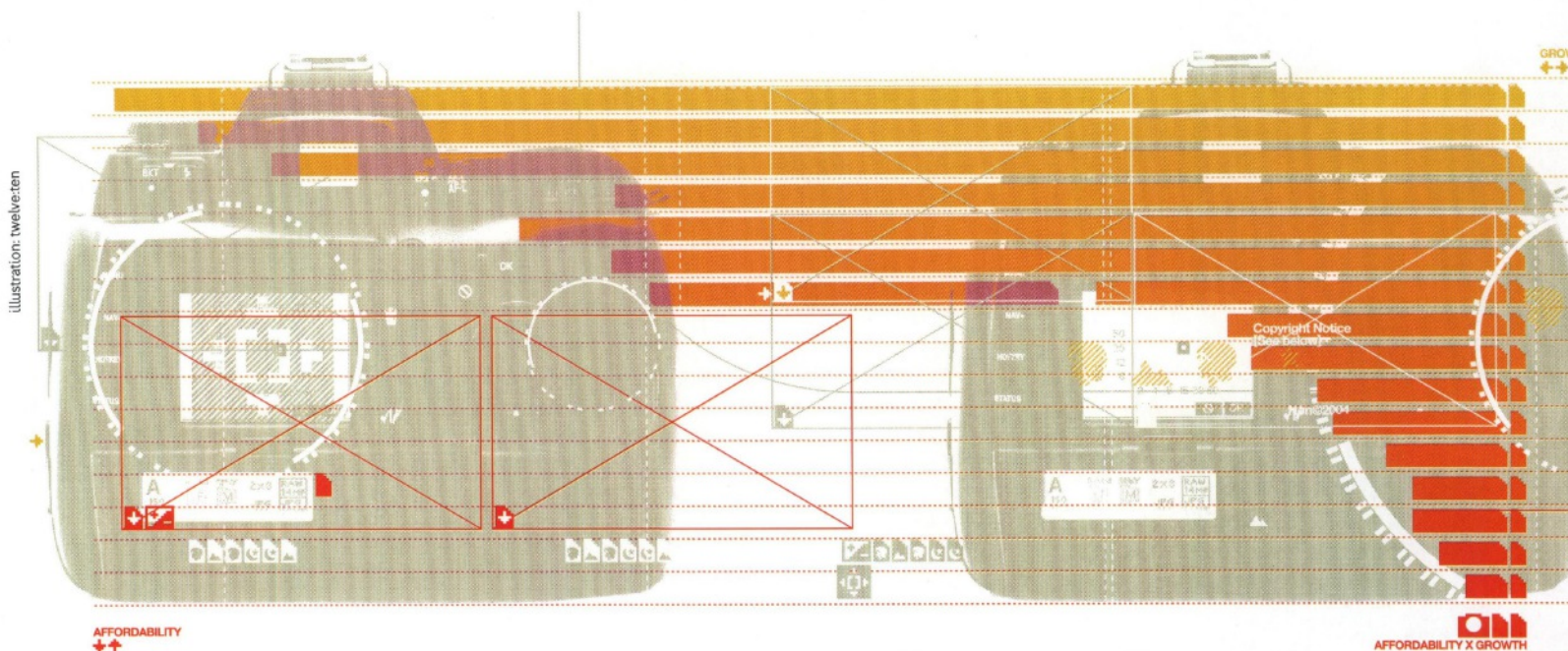
Just because you can doesn't mean you should. A news portal doesn't need a 5MB intro, even if it is geared towards a broadband audience.

4 Fast but not that fast

Broadband might be fast, but in general it still means between 500KB/s and 1MB/s. Sites with whacking great 50MB Quicktime movies are still going to involve a wait.

5 Usability

If they've got broadband they'll be experts – they'll figure it out. Now broadband is widespread, you can't guarantee who'll be using the site or what their requirements are.



Royalty-free images fight back

Royalty-free stock images are in the ascendant, shrugging off their reputation for low quality and the suppliers of more expensive rights-managed material are on guard, says **Dom Hall**

Although once seen as the poor relation of the stock image market and traditionally blighted by quality issues and unnaturalistic subject matter, the demand for rights free (RF) images has exploded over the past two years. As a result, the lower cost alternative to expensive and complicated-to-use rights managed (RM) photos has become of tantamount importance to both creative users and the companies who supply them – and accountants everywhere!

Design industry analyst TrendWatch has tracked the use of RF images in the US and found that typical use among creative, design and production companies has risen over the last year from an average of three images to up to seven images a month – a trend the company's director Jim Whittington believes has been mirrored in the UK and Europe.

"There has historically been a higher demand for royalty-free images over licensed images because of costs. Not all projects require licensed-use stock, nor do all projects have a budget for purchasing licensed-use images," he says.

"Many RF images are used on websites where, due of copyright control, it doesn't make sense to use licensed images. Another reason, perhaps, is that there are simply more royalty-free images available. From what we see, the over-riding reasons are price first and availability second," says Whittington.

Louis Ingram, director of UK-based stock library Ingram Publishing backs this up. He says the industry will open up the use of stock images, particularly RF, to smaller companies

previously deterred by costs and contracts.

"We feel the market is opening up to smaller and smaller graphic design agencies. There is no way a small business would spend \$300 on a single image when royalty-free images are cheaper and can be used again and again," he says.

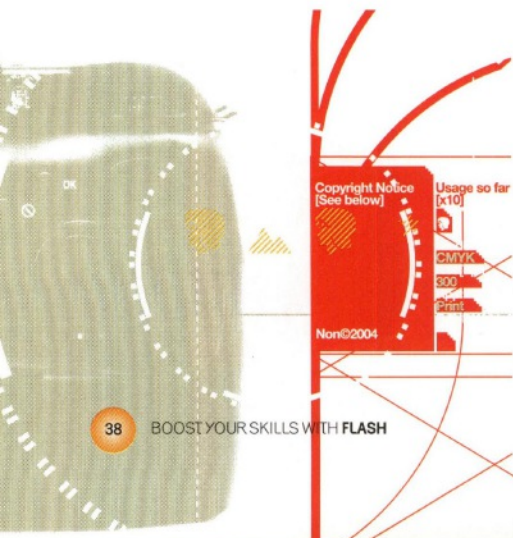
Price wars

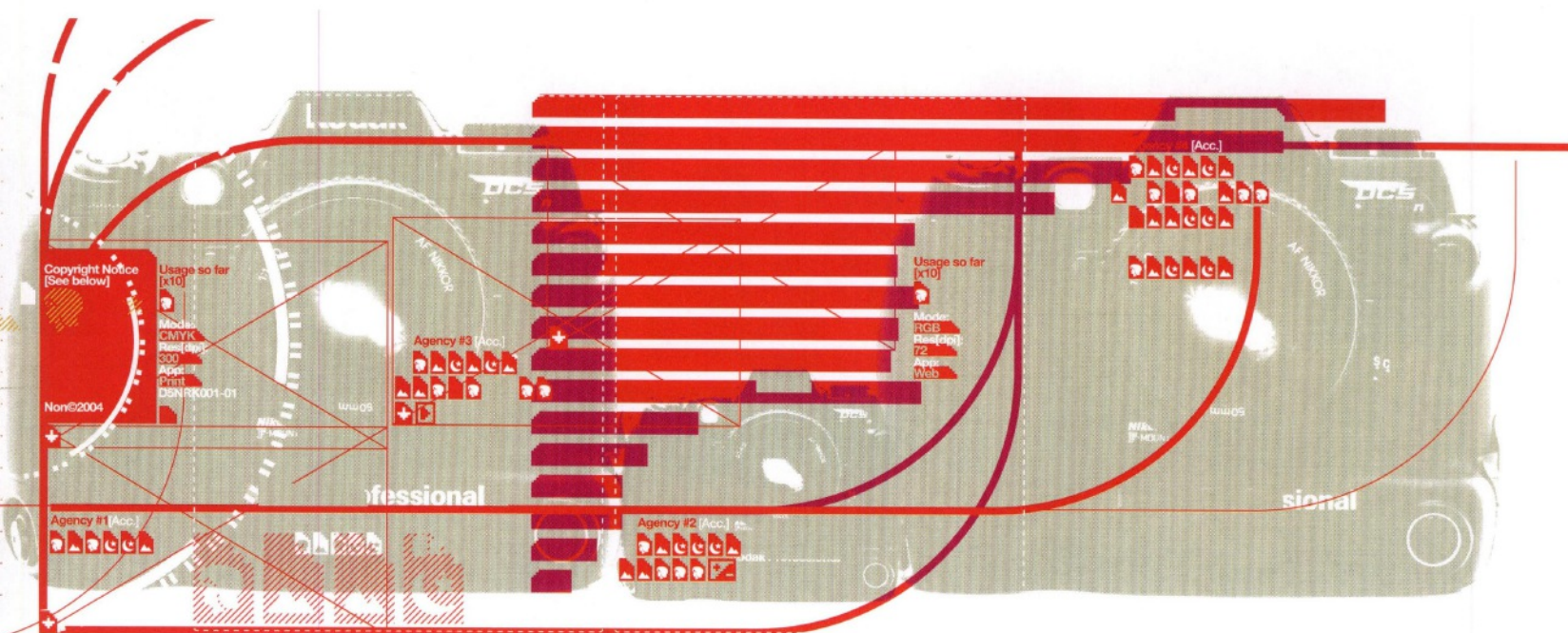
But is it really as straightforward an issue as price? After all when it comes to creative projects, design agencies typically strive to produce the best work they can using quality source material, regardless of cost.

Not so, says Ben Brown, creative director at design group The Creation Agency, who says clients are often driven purely by price.

"We use RF images because of the price – it's cheaper and image use is easier to manage. We try and push clients towards RM when we can because the quality is higher.

"Quality was always a problem with RF – the photography was usually of cheesy business shots and compositions, but that's changing and we are having to re-programme





"The extended use of RF has led most big producers in the industry to switch emphasis from RM to RF production in order to service the high demand, which has in turn led to fewer quality RM images being produced," Glenister argues, adding that he believes the

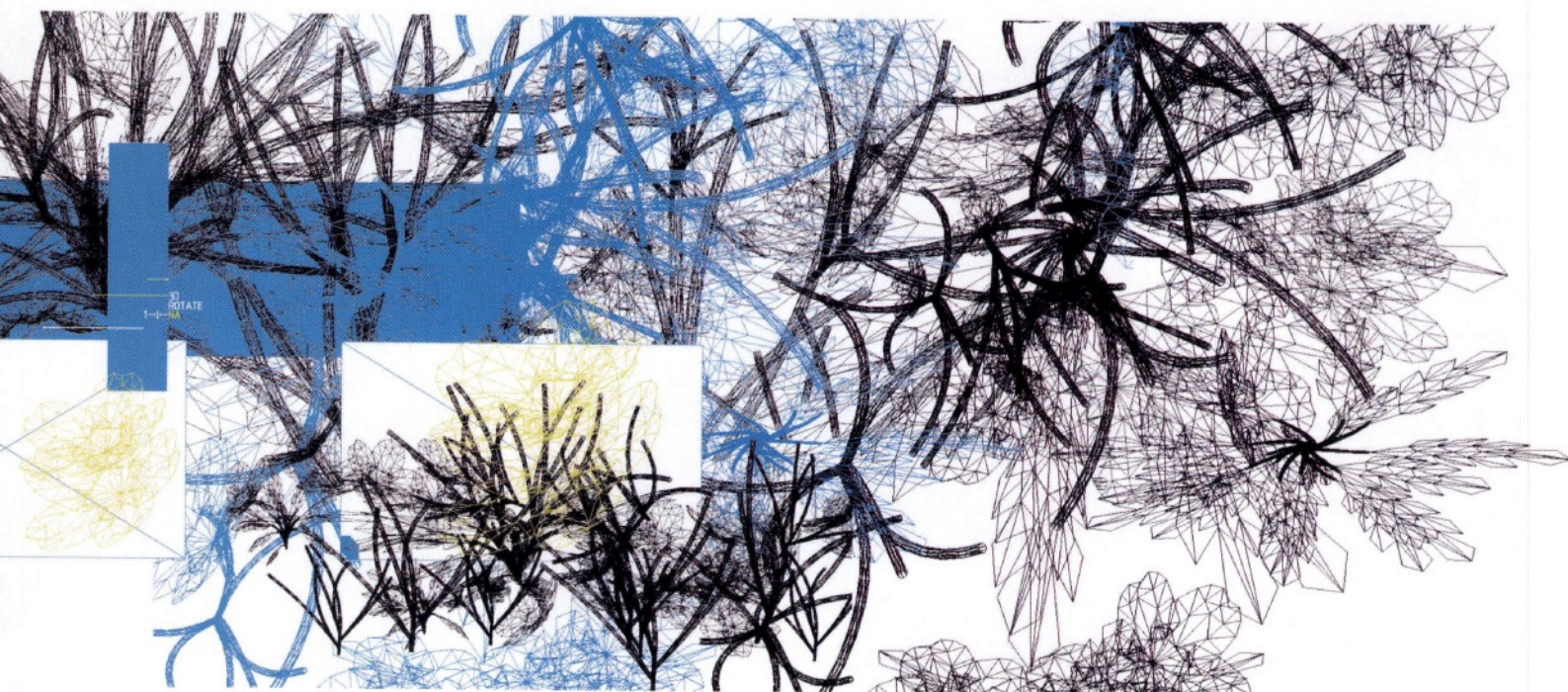
"[Producers of] RM images are worried, and so they should be. The threat of RF is not going away"

"[Producers of] RM images are worried, and so they should be. The threat of RF is growing and is not about to go away. There may be a coming together of the two models at some point — but the logistics of how this will happen are the underlying reason why it hasn't happened so far. RF producers are having to spend more and more money on production, and returns per image are a lot lower than they were last year," he argues.

"We are watching new licensing developments and expect to see more use of new models. There are experiments offering a broader use of images with a rights-managed license that has the same ease of use as a royalty-free image but with some limitations. We're also seeing more use of a subscription-based approach," he says.

For an intelligent critique of digital rights management, download a PDF of the manifesto written by respected tech author Cory Doctorow from the last link below.

<http://pro.corbis.com>
www.image100.co.uk
www.creationagency.co.uk
www.trendwatch.com
www.changethis.com/4.DRM



if you make your living as a graphics professional, \$2,000 at the upper end isn't too much," says Ruffo.

"It's also no more difficult to master than any other 3D package. A few years ago, there was no 3D in television ads; now you can't turn on the TV without seeing it. The same thing is happening in the professional space for graphics."

Over at Solid, director Gavin Lindsay explains that his company has been using Maya for the last seven years to create high-resolution computer-generated imagery for

commission from cosmetics manufacturer Estée Lauder. The agency had the rather pressing problem of how to present an image of a Clinique moisturising product that wasn't actually available to photograph.

"The moisturiser was still in development but we were able to create a poster as it would appear," Reproline's director, Andreas Achter, explains. "Being able to have posters ready to go before the product is even finished reduced time-to-market for our clients and it's a compelling service to be able to offer them."

The death of film

While plenty of people are clearly welcoming the broader use of 3D design packages, the development is not to everyone's taste — especially the traditional image powerhouses of the creative industry: the photographers.

London-based photographer Charlie Gray believes "a good percentage of photographers are concerned by the trend" — the advertising industry is already well aware of the many and varied ways 3D images can save money.

"This even stretches as far as live models and people shots. So many advertising campaigns are created using computer-generated people or characters in product shots. That way, they avoid expenses such as worldwide buyouts, which can be extremely fruitful to models and photographers alike," Gray explains.

"This could be damaging in the long run," he adds. "But, to be realistic, people were predicting the end of film when digital cameras came into the frame."

For Warren Palmer, a Hampshire-based photographer who specialises in corporate and advertising images, the trend for computer-generated images is just that — a trend that will eventually run out of steam. He reckons the similar look and feel of 3D images in advertising will eventually force agencies to re-think and seek out something completely different.

"Ad agencies tend to jump on the bandwagon. Once CG images reach saturation point, the circle will go back to the start again," he says. For the time being, though, it seems that we need to prepare ourselves for an onslaught of 3D imagery — as well as increasing commercial pressures that will force graphic artists to decide which side of the fence they're sitting on. ■

INFO

www.alias.com
www.solid.co.uk
www.reproline.de
www.charlie-gray.com
www.wjpphoto.com

"You just can't achieve the kind of things we can using Maya in the real world. That's why people come to us"

advertising. For Solid, the benefits are simple, the program helps the company to provide any kind of image without real-life restrictions and at relatively low cost. This is something clearly demonstrated in commissions the design firm has carried out for a number of high-profile clients, including Perrier, Land Rover and Kellogg.

"You can't achieve, in the real world the kinds of things we can using Maya," he says.

3D software helped German design agency ReproLine to meet a recent



The mere idea of things going wrong on a web design project can send shivers down your spine. How do you cope when things go wrong and how can you prevent them from happening in the first place? **Oliver Lindberg** quizzed designers about their worst fears and secret solutions...



Nightmare scenario 1

Lela Kitley, creative developer at digital communication company, Reading Room (www.readingroom.com), says, "The biggest nightmare must be clicking that upload button when you're making a site live. Always worrying that you've missed something somewhere, that the live server is going to eat your site and that it'll take hours to fix, while the client is phoning up and asking why their site isn't working. The best way of preventing any disasters is to copy any existing files from the live site and keep them somewhere safe. Then, if the worst happens, you can put them back up while you work out what the problem is."

Argh... the site doesn't work!



Nightmare scenario 2

Work on the project has begun and the design is already complete. Then you find out that the requirements have changed. John Penrose, technical director at Graphico New Media (www.graphico.co.uk), says, "This could be due to a misunderstanding, or a change outside of your control, such as the client's business model changing. As you become more experienced, you learn to spot areas of risk in the scoping stage of a project. By asking the right questions, you can make sure that everyone has the same expectations of what will be produced, and what it will and won't do. Ensuring that this is documented and signed off by the client ensures that misunderstandings are minimised."

What do you do if you can't prevent the project from changing halfway through? "Having a change control procedure ensures that all side effects of the proposed change are considered and the client is made aware of any cost implications," John explains. "Careful planning in the technical design stage should take account of the fact that everything changes. Finally, making sure that documentation such as technical specifications and user manuals are updated to reflect the changes is vital to ensure that any future changes are based on up-to-date specifications."

You've changed your mind!?



Nightmare scenario 3

Naturally clients can be a nightmare too, but dealing with them is your job. You have to be prepared when they want you to knock down the price, change the look of the site at short notice or get too involved in the design process. Kyle Millard, senior account executive at web site design agency KD Web (www.kdweb.co.uk), says the company's worst nightmare is having a fussy client with the worst design taste possible. "The way we now deal with such clients is by getting a very detailed brief to start with," Kyle explains. "We ask a series of questions designed to find out exactly what the client wants the web site to achieve. We also ask the client to provide a list of web sites that they like the look of so that we know their personal taste. We then state the number of hours and design changes we are willing to do on the contract."

Jonty Balcombe, director at web design agency, Peppercorn (www.peppercorn.co.uk), thinks it might be a good idea to get your design signed off before you start working on the code for the site. If you also ensure the client sees the site being built on a staging server, there shouldn't be any last minute shocks. If the client then changes his mind a week before launch, try to identify which elements he doesn't like and assess whether they can be changed easily. If this is impossible, try to negotiate additional fees and more time.

Darn those fussy clients!



Nightmare scenario 4

The feeling you get when you realise that you've lost important data is not pleasant. Helplessness, shock, your life is going to end... "I think it must have happened to most web designers at least once," says Tim Diacon of creative consultancy, Crush (www.crushed.co.uk). "A simple click on the wrong button in an FTP program or saving over the wrong file can undo a lot of work." However, this nightmare can be avoided easily. You know it might happen, so just take a few precautions: maintain regular backups of everything and save different versions so you can revert back to any at any time. Also, ensure that you have good terms and conditions of trade signed by the client before the start of the project. This should cover you in such an event. "There is no right way of safeguarding against this kind of mistake and perhaps it's best to come up with your own system," Tim says. Remote servers can be synched to local files or duplicate folders could be made. You can be sure, however, if you don't take some kind of precaution you will slip up."

Whatever you do, don't lose your head. Guy Moorhouse of freshlysqueezed (www.freshly-squeezed.com) advises "keep calm and apply the following maxim: 'If in doubt, restart.'"

Help! I've lost my data





Nightmare scenario 5

Halfway through a project you realise that you have underquoted and could miss out on a big bunch of money. It can happen so easily and, when it does, the consequences can be significant

"There's very little you can do if you are under contract,

except perhaps brainstorm a cheaper solution that the

client will be just as happy with," says

Peppercorn's Jonty Balcombe. Therefore,

you should take precautions to

avoid being in an uncomfortable

situation like this. "Be as

specific as possible when

quoting for work," Jonty

recommends. "Do some demos of

any possible problem areas before

agreeing prices with the client or,

alternatively, never give fixed

quotes, just have hourly rates."



Our problem solvers



Top row: Lela Kitley, John Penrose, Kyle Millard, Jonty Balcombe. Middle row: Ronnie Wright, Paul Newman, Tim Diacon. Bottom row: Trevor Spink, Louis Whurr, Guy Moorhouse

Nightmare scenario 6

Finally finishing a project is a great feeling but it might not be over just yet. Imagine you have one hour to upload the files and then a server goes down. Trevor Spink, managing director and chief technical officer at web design and web software development company, Eonic (www.eonic.co.uk), says "You always need to rely on a hosting provider, and when they have problems, the ramifications are huge. To have 50 client sites disappear can be chaos. We once inherited a server from another company that was virus infected and we had to rebuild it overnight – it was hosting 50 web sites all running bespoke ASP code with file, DNS paths hard coded throughout."

Tim Diacon of Crush knows the scenario, too. "It really is a situation in which you can do absolutely nothing to help yourself," he says. "The only way to stop it happening is to make sure you get your web space from a good company. **It's always best to get a professional recommendation as opposed to searching for cheap web space on the internet. You pay for what you get.**"

Jonty Balcombe agrees. "You should employ the services of a very reliable, experienced hosting company with good technical support," he comments. If the server still makes problems, keep the client fully informed on the fixing progress and how long you estimate the site will be down. Also ensure you always keep careful backups and have a disaster recovery plan. If your client makes updates to the site, make sure you regularly download the site and back it up.

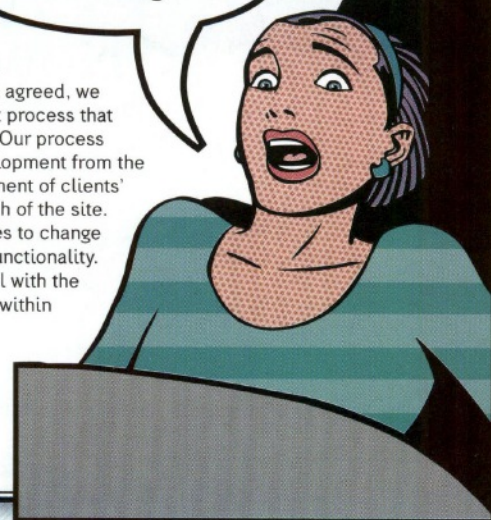
...Freakin' server!



Nightmare scenario 7

The project is underway and you realise it hasn't been scoped thoroughly. This can make your life much harder than it needs to be. First, you have to define the elements that are missing. Then you need to get back to the client with a list of questions, while time is running out and you suddenly find yourself under a lot of pressure. To avoid this, **you should make sure that your client provides all necessary information before the start of the project.** At web design agency, guava (www.guava.co.uk), the account management and development teams follow a procedure to establish the business objective as well as the functionality requirements, content, restrictions and much more. "After the information has been obtained, we provide a full project specification that both the client and guava agree to," says guava's director, Paul Newman. "When this has been agreed, we work through our development process that highlights project milestones. Our process covers all aspects of the development from the initial creative to the management of clients' change requests and the launch of the site. On occasion, the client decides to change the creative, or wants to add functionality. Our process enables us to deal with the request by simply including it within the project or the re-specification if the request is out of scope."

I've got this all wrong...



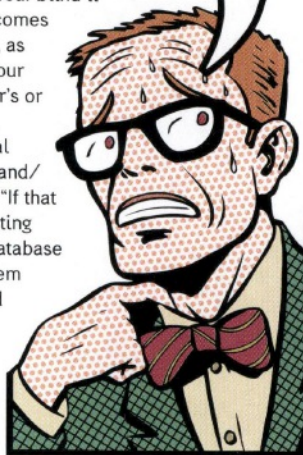
Nightmare scenario 8

The biggest fear of Louis Whurr, client site developer at international communications agency, Worth Communications (www.worthcommunications.com), is 'something known as a technical gremlin (technicus gremlinus)'.

"That's the kind of nightmare that involves something not working normally on random machines and for random reasons and offering no clues as to why," he explains. "Normally, it's being resolved at silly o'clock in the morning, followed by not so much giving yourself a kicking, but more a brutal beating in the nearest pub car park for being so stupid. Currently, I have a web site, which disappears, and the DNS servers swear blind it doesn't exist. Then, inexplicably, it comes back. Not a happy situation to be in, as the uninitiated clients assume it's your fault and not the web space provider's or the domain provider's."

"Normally, it's best to try the usual suspects first, such as broken links and/or a server going down," Louis says. "If that all checks out, then I move on to testing other things like broken code or a database error. Finally, I write down the problem concisely, using my best English and email a bunch of mates that have 'ologies in clever things. If that doesn't work, it's normally something quite crucial and the most economic thing to do is rebuild it. I then take careful note of exactly what I'm doing."

Nooo... not the technical gremlin!



Nightmare scenario 9

Getting hold of a domain name that has been registered previously can be tricky. Often the registrant has completely vanished, which makes it quite a challenge to get the domain released. DNS transfers can mean a lot of work. "They never go smoothly," says Eonic's Trevor Spink. "You can guarantee that your client got some two-bit bedroom web designer to register their domain name. They have now lost touch — hence you winning the job. Getting hold of a domain when it has been registered to a third party can be a nightmare. One site I had took nine months to get the previous registrant to release the domain. The address at Companies House didn't exist, accounts hadn't been returned for years. When I finally tracked them down, they kept promising to send the faxes off to confirm the transfer but never did. Eventually I had to turn up on their doorstep and stand over them whilst they signed them and faxed them off to Nominet." Ronnie Wright of graphic design studio Wright

DZN (www.wrightdzn.co.uk) had a similar experience. "I recently

took on a job to upgrade and host a site on our servers," he says. "In order to do this, I have to request the current web developers that are hosting

the web site to change the DNS to our servers. This is where the problem started. It took me three weeks to get hold of them and once I did, they made false promises and tried everything in the book to destroy my contract with the client. All my hard work to get the client to transfer the business my way was nearly knocked for six when they contacted the client and tried to undermine my service and offer a better deal. Luckily, I had a signed contract and the client's faith, as I had spent some time explaining how the web site could be made better."

Those devilish DNS transfers!



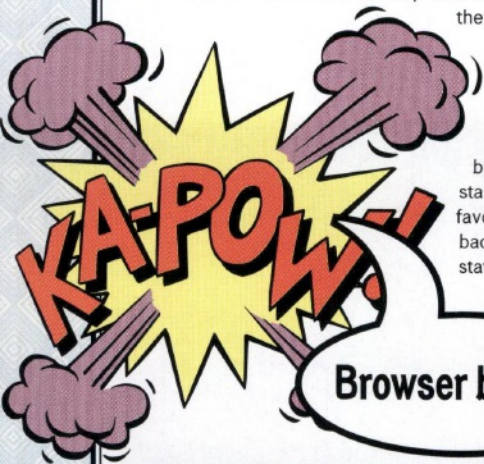
Nightmare scenario 10

There's no doubt about it: cross browser compatibility is important. Your client will want as many people as possible to be able to access their site but it can also give you a real headache. "My worst nightmare is someone who asks for a database driven web site or content management site that will work cross-platform, on all legacy browsers including IE5 for the Apple Mac and Netscape 4 on the PC," sighs Sean Flynn, responsible for ecommerce and ebusiness development at Hangar17 New Media (www.hangar17.net). "Have they never heard of IE6 or Firefox or Netscape 7?"

I just wish people would upgrade their browsers as often as they upgrade their mobile phones. A new browser doesn't even cost you a penny, especially if you get .net magazine!

So what do you do if your client insists on old browsers? "Think accessibility and web standards," Sean advises. "Some laws do have a favourable side and, whilst you're there, do some background searching and look through the web stats to see if you can't illustrate why old browsers just don't get — or shouldn't get used anymore. With spyware and anti-pop-up ad blockers, why would anyone ever want to use a security-flawed browser? Otherwise, always be realistic with your client."

Browser brawl!



DREAM SOLUTIONS

Ten tips to avoid the worst nightmares

- 1 Have a signed contract.
- 2 Plan your project carefully before you start and try to find out as much as possible about it.
- 3 Have a thoroughly documented and detailed written brief.
- 4 Get your client to sign off both the brief and the design before starting the code.
- 5 Get to know your client and involve him/her as much as possible in the development process to avoid surprises.
- 6 Have an effective procedure to take care of client changes. Keep a printed record of any change requests.
- 7 Back up and save your files as often as possible.
- 8 Have a disaster recovery plan.
- 9 Go for a very reliable, experienced hosting company with good technical support.
- 10 Try to finish your project 48 hours ahead of schedule.



Make Your Site Standard

Want your sites to work on any platform, be portable and predictable? It's time to bin browser specific code and follow the letter of the law.

Karl Hodge shows you how to apply the latest web standards to your site. Illustration: Sean Rodwell

For the World Wide Web, 1992 to 1995 was a crucial period. The folks who created and nurtured it, Tim Berners-Lee and his team at the University of Geneva, looked on altruistically as their creation went global. Waiving all royalties and intellectual property rights on the protocols they'd come up with, those years saw rapid development as web servers spread through the universities of Europe and the USA. By 1994, a few key figures, including Berners-Lee, saw the need to take back control of those protocols and the World Wide Web Consortium was formed. Its stated aims were to make sure that the web would remain accessible and open to everyone. As history shows, the World Wide Web Consortium, or W3C as it's affectionately known, was convened at exactly the right time.

The same year the W3C was formed saw the release of the first browser with proprietary additions to HTML; Netscape Navigator. Non-standard extensions to HTML pioneered by early versions of this browser included the `<font>` tag, the horizontal rule and the `<blink>` tag. Many Netscape extensions were absorbed into W3C endorsed versions of the language,

becoming standards in their own right — but the practice of including proprietary extensions set a precedent that was difficult to back away from. The situation really became a problem when Netscape and Microsoft began to add competing tags and attributes to their browsers in the late 90s. Now we're left with a soup of browser specific extensions, attributes and syntax... which is where web standards come in.

The most persuasive commercial argument for following standards is that they make sure your pages work whatever browser your punters are using, whatever platform they're on.



are actually there to make things easier for you, not harder. Once you start to apply them, your pages will work more consistently, your code will be more portable and it will be built to last.

Making Up Mark-Up

Web page mark-up languages are at the core of the standards debate. Over time the W3C has evolved a series of standards around the web's default mark-up language, HTML. The most recent version of HTML proper was 4.01. When building static pages, this specification should be your baseline reference. Bluntly, if you're coding in HTML and a tag or method isn't supported in 4.01, don't use it. You can find a very full specification of the standard at www.w3.org/TR/REC-html40.

One area of the spec you should keep an eye out for are deprecated elements. The usual reason a tag is deprecated: there's a better, more consistent method of achieving the same result available. In HTML 4.01 the W3C pushes alternate methods of text formatting using CSS, for example. Stalwart tags like `<font>`, `<u>` and `<strike>` are deprecated, along with almost every attribute that changes the appearance of or formats an inline element.

Though use of these elements is discouraged, if you have a well-formed HTML 4.01 document that includes them, it will still work. How long that'll be true for is another matter. Just around the corner are some radical updates to the default mark-up language for the web. The most far-reaching of all is the shift from HTML, a fixed mark-up language for browsers, to XML, a global standard for the creation of mark-up languages.

Documents defined

XML separates the rules of a language from the browser, placing it in a separate text document called a Document Type Definition or DTD. This makes browsers more flexible, placing power in the hands of developers and enabling them to create bespoke languages for specific purposes. For static web development it makes it much easier to ensure that your pages are consistent on different platforms. Instead of using the web browser's built in rules, you can make the client software use your definition.

So that your browser knows what DTD you're using and can render your code correctly, you must declare its type at the top of your page before anything else. What isn't well known is that this line of code, known as a `!DOCTYPE` declaration, has been part of the HTML standard for some time. In their wisdom, the big browser manufacturers simply ignored it.

Increasingly this extends to platforms that aren't desktop computers; PDAs, for example and other mobile devices. And, of course, open access for all is built into W3C standards — so following them is an essential step towards complying with UK and European accessibility regulations.

But where do you start? There are standards for different flavours of HTML, client side scripting rules, and at least two kinds of CSS to consider. That's before you start to think about browser and platform differences.

Fear not. It may be hard to believe, but all these standards



Microsoft was the first to introduce compliance with !DOCTYPE declarations in the year 2000.

In modern browsers, a !DOCTYPE definition renders pages in a way that complies with the standards of the HTML version that's declared. If there's no !DOCTYPE declaration, or a pre 4.01 !DOCTYPE declaration, your browser defaults to quirks mode. In fact, if you've had trouble with tables or formatting that behaves in odd, unexpected ways on different browsers, the chances are that it's because you missed the !DOCTYPE definition at the very start of the code, before the opening `<html>` tag and your browser is using this special legacy mode.

HTML 4.01 was the first standard version of HTML to have an external DTD, beginning the march towards XML compatibility. In fact, it has three DTDs. There's the Strict DTD that complies with the absolute letter of the HTML 4.01 standard. This version excludes all the deprecated elements we mentioned before — so there are no presentation elements at all and formatting must be achieved using style sheets. It also excludes the legacy `<iframe>` tag, its functionality replaced by the `<object>` tag.

The most widely used DTD is the Transitional version. This includes the deprecated elements, allowing for a grace period for authors to get used to style sheets — but it also incorporates all the newest 4.01 functionality. Finally, there's the Frameset version, which is identical to the Transitional version, except that there's built in support for frames. The !DOCTYPE declaration to define your page as HTML 4.01, Transitional is as follows:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

You'll find a list of DTDs at www.w3.org/QA/2002/04/valid-dtd-list.html. The future of web standards is looking predictable. XHTML 1.1 jettisons deprecated presentational tags — but the next version goes further. In the current XHTML 2.0 draft, published in August 2002, backwards compatibility has disappeared. The `<b>` and `<i>` tags will go, and even the `<img>` tag will be superseded by the more general `<object>` tag. It will be a simpler, more intuitive mark-up language, but one that will take some getting used to if you're still using HTML to format text and tables to layout pages.

XML marks the spot

Sticking to standards now will make your sites better in the present and ready for anything in the future. A full XML-based successor to HTML already exists. XHTML

JAVASCRIPT, JSCRIPT OR ECMASCRIPT? THE ONGOING DEVELOPMENT

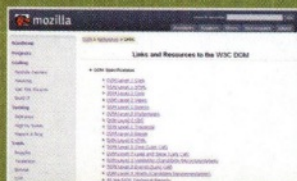
There was one kind of JavaScript at first, created by Netscape as a standard for client-side interaction. IE started to support it with version 2.5, but added its own extensions to the language and called it Jscript. With the release of IE 4.0 came a major challenge to Netscape with the introduction of DHTML, a souped-up Jscript that enabled developers to reference any page element and change it on the fly.

Still, there are applicable standards and the strength of the Mozilla base and standards compliant Firefox as a browser is making them more and more

difficult to ignore. JavaScript is now overseen by the European Computer Manufacturers Association (www.ecma-international.org) — in charge of several computing standards. ECMA compliant JavaScript is a simplified, more consistent and more generic version of the original. See www.mozilla.org/js/language.

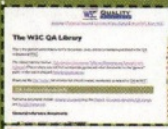
That's only part of the tale: after the DHTML debacle, the W3C were prompted to formulate their own standard for referring to and accessing the structure of web pages. Their recommendation, Document Object Model level 2, is a standard structure for web pages that are fairly well supported in Firefox, but less well implemented in IE6. For now, using

DOM level 1 object methods and ECMA syntax is the standards compliant compromise for writing JavaScript. Find out more at www.w3.org/DOM.

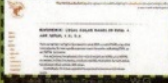


With typical altruism, Mozilla provides references to both its own and other browsers shortcomings when dealing with the W3C's DOM level 2

TEN USEFUL SITES TO VISIT TO LEARN MORE



The World Wide Web Consortium's site is so comprehensive and in-depth that it's difficult to know where to start. We'd recommend the Education and Outreach section of the Quality Assurance Library as the best jumping off point — www.w3.org/QA/Library



Next to the W3C, The Web Standards Project is the web's best resource on creating pages that meet agreed specifications — www.webstandards.org



The Web Standards Group About Web Standards section is a must-read overview of the issues involved in making standards based sites — webstandardsgroup.org



MACCAWS (Making A Commercial Case For Web Standards) formed to persuade ecommerce and business sites to embrace cross-platform, W3C endorsed code — www.maccaws.org



A brilliant reference to the structure of HTML and XHTML documents, Index DOT HTML is an indispensable bookmark — www.blooberry.com/indexdot/html



It's difficult to find good guides to the W3C's Document Object Model, which is why this Indispensable one has such a long URL — zvon.org/xxl/DOM2reference/Output/index.html



CSS2 exists as a real, documented standard, but Internet Explorer only supports bits of it. For now it's best to code with CSS1 in mind — and you'll find a full reference as the Web Design Group's site — www.htmlhelp.com/reference/css



The official, browser neutral standard for ECMAScript can be found as a PDF file deep within www.ecma-international.org — tinyurl.com/2ltdr



As standards are constantly changing, a blog that details and discusses the latest projected innovations in web design is exactly what you need — www.alistapart.com



Quirks Mode is an invaluable resource when troubleshooting sites. With sections on CSS, mark-up, browsers and more — the author attributes on multiple platforms in quirks and strict browser modes, so you'll always know what to expect — www.quirksmode.org

1.0 is a replica of the previous HTML standard in XML, and the standard that you, as a developer, should most probably be building your pages in. The elements within the language are virtually identical to 4.01 — which is why we spent so much time explaining how that works first. There are changes, though, to the structure and formatting of the mark-up itself, which is now subject to stricter XML guidelines.

To begin with, XHTML tags are always formatted in lower case, like this:

```
<title>Standards Compliant Page</title>
```

In HTML the convention was to write your tags in upper case, but it didn't actually matter to the web browser whether you did or didn't. Also, there's a standard stating that mark-up should be well-formed, meaning that it should be written in a consistent way. All paired tags have to be closed in XHTML — so where you might have left a paragraph tag open, like this:

```
<P>Here's a paragraph.
```

You must now always place an end tag after the content:

```
<p>Here's a standards compliant paragraph</p>
```

Also, you should take care when nesting tags, making sure they don't overlap. There's another change that's new to XHTML mark-up: so called empty elements, like the `<br>` tag, must now be closed. This is achieved with a terminating slash before the last bracket: `<br />` instead of `<br>`

This rule applies to any tag that doesn't have a close tag in the original HTML specification, including the `<img>` tag:

```

```

A standard document must — of course — have head, title, and body structural elements. A frameset document must have the head, title, and frameset structural elements.

THE FIVE MOST COMMON VALIDATION PROBLEMS AVOID THEM IF YOU CAN

1 HTML mark-up within `<script>` tags may be read incorrectly as part of the HTML script. You can either avoid using HTML within scripts, or place a backslash in the tag to prevent it being parsed as mark-up, like this:

```
<script type="text/
javascript">
<!--
document.write("</H1>");
// -->
</script>
```

2 In a DOCTYPE declaration, the formal public identifier, the string that declares the actual document type, is case sensitive — it won't work if you enter it in lower case. Refer to www.w3.org/QA/2002/04/Web-Quality for the exact declaration required.

3 Poor nesting of elements will lead to validation errors, and errors in some browsers. If your code looks like this:

```
<p><a href="www.netmag.co.uk">
Click</p></a>
```

It should look like this:

```
<p><a href="www.netmag.co.uk">
Click</a></p>
```

4 The W3C's validator chokes on some special characters, predominantly those reserved for use in mark-up. One to commonly look out for is the ampersand or `&` — especially in URLs. Replace it with the ASCII equivalent `&`.

5 Missing alt tags are a common standards error across all document types, as are missing quotes around HTML attributes. Double-check before uploading your pages.

In addition to these more tightly enforced formatting conventions, there are a couple of key differences when declaring an XHTML IDCTYPE. An XHTML document must include an XML namespace definition `xmlns` within the opening `<html>` tag. Although it's not absolutely necessary, XHTML authors are strongly encouraged to begin XHTML documents with an XML declaration too. However, Internet Explorer 6 defaults to quirks mode if you do this, even if you have the right IDCTYPE declared — so it may be best to make an executive decision, and leave this out. Put all that together and you get something like this:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/
xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xml:lang="en" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/
html; charset=iso-8859-1" />
<title>XHTML 1.0 Compliant Page</title>
</head>
<body>
<p>Content Here</p>
</body>
</html>
```

Like HTML 4.01 — there are Strict, Transitional and Frameset versions of the XHTML 1.0. Which of these standards and variations do you choose to build your site with? For the sake of future compatibility we'd recommend that you build any new sites using XHTML 1.0 specifications — and aim for Strict DTD compliance if you can. If you have an older, HTML coded site, then use the HTML 4.01 Transitional standard to bring it into line.

It should also be noted that while the most widely used current version of XHTML is 1.0, there is a newer version: XHTML 1.1. This standard is a modular rendition of XHTML 1.0, based on the Strict DTD, jettisoning all deprecated, presentational elements. This strictness is part of the W3C's ongoing effort to promote the separation of content and form — and to achieve standards compliance that's what you should be aiming for. You can boil down the message to one sentence: define your static content with well-formed XHTML and format it with stylesheets.

Putting on the style

On the W3C's perfect web, external stylesheets would always control page layout. While embedded CSS layouts and even table layouts remain part of the XHTML standard, you can work towards dividing content and formatting by using external stylesheets for formatting at least. The stricter versions of HTML and XHTML leave you little choice in the matter.

CSS standards currently offer many ways to include styles in documents. Common sense dictates that you should be consistent when using those methods: try not to mix external styles with embedded stylesheets, or use an inline style attribute to format text when you also have a stylesheet declaration at the top of your document. It's permissible, and usual, to do content formatting with an embedded or external stylesheet and layout with inline styles within your document, though. Of course, mixing CSS styles with `<font>` and other tags — like `<b>` for bold text and `<i>` for italics — is not the way a progressive designer would code.

Like HTML and XHTML, there are well-defined standards for CSS maintained by the W3C — but uptake for those standards has been slow. The CSS2 specification has been around since 1998, but support for it remains patchy. If we're to meet the W3C's Holy Grail of pages that separate layout and content, we have to fall back on the CSS1 standard — or test CSS2 elements in target browsers on a case by case basis.

When you've finished building your XHTML 1.0 strict, CSS formatted pages, there's one last step to confirm your compliance with web standards: validation. The W3C has free validation tools for both web pages and style sheets — you'll find them at validator.w3.org and jigsaw.w3.org/css-validator respectively. Helpfully, if any elements of your code have standards-based errors in them, these tools suggest how you might fix them — which makes them a great first port of call if you're planning to overhaul an old site, too. A visit to these validators is essential whether you've hand-coded your pages or used a drag and drop authoring tool. Even largely standards compliant tools like Dreamweaver and Adobe GoLive may allow non-standard mark-up, confining their error checking to browser support rather than W3C standards. ■

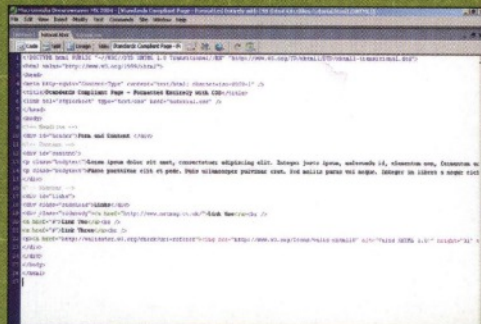
IMAGE STANDARDS

THOUGH THERE ARE NO STANDARDS THAT DEFINE WHICH GRAPHIC FORMATS YOU CAN AND CAN'T USE, IN PRACTICE SOME ARE MORE SUITABLE FOR CERTAIN JOBS THAN OTHERS

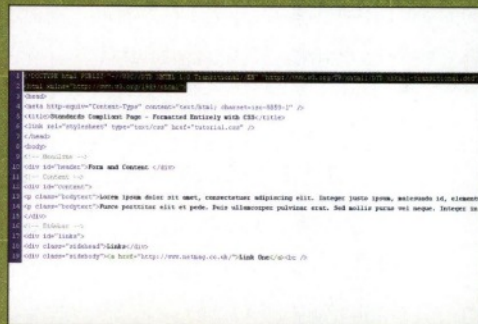
Format	GIF (Graphics Interchange Format)	JPEG (Joint Photographic Experts Group)	PNG (Portable Network Graphics)	BMP (Windows Bitmap File)	SVG (Scalable Vector Graphics)
Specifications	8 bit, 256 colour fixed palette, transparency, animation, image compression	8 bit greyscale, 24 bit colour, 16.7 million colours, lossy compression	8 to 64 bit colour, true colour, lossless image compression, 256 transparency levels	1 to 24 bit colour, 16.7 million colours, image compression	W3C endorsed, XML based vector graphics format for the web
Browser Support	Global (IE, Firefox, Mozilla, Opera, Safari)	Global (IE, Firefox, Mozilla, Opera, Safari)	Partial (IE, Firefox, Mozilla, Opera, Safari)	Internet Explorer only	Partial (IE, Mozilla, Firefox)
When to use	GIF's limited palette makes it ideal for simple graphics with areas of flat colour	For photographic images or images containing multiple, banded colour transitions	Intended as a successor to both GIF and JPEG, PNG files currently replace GIFs with most success — though alpha transparency is not yet supported in IE	A universal bitmap image format. Not suitable for general use on the web	Hold your fire. At the time of writing, SVG was only supported in IE through the Adobe SVG Viewer plug-in. Though new builds of Firefox and Mozilla include built-in SVG, the feature's not projected to be ready for official release until March 2005

TUTORIAL CREATING COMPLIANT PAGES WITH STYLE SHEETS

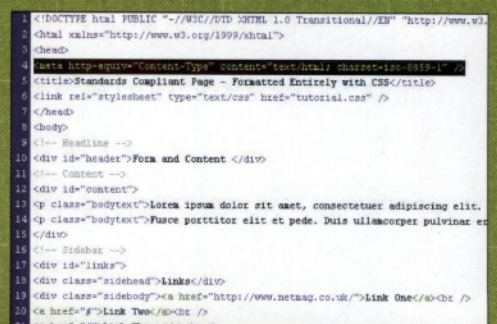
The Holy Grail of web standards is separating form from content. Here we look at the anatomy of a simple, standards compliant web page that's completely laid out with stylesheets



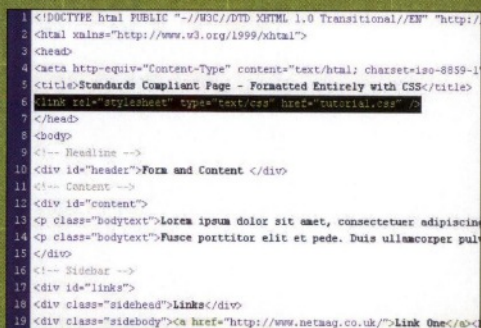
Open the files tutorial.html and tutorial.css in your favourite web-authoring tool. We're using Dreamweaver for its help with standards compliance and support for CSS. We'll look at the HTML file first.



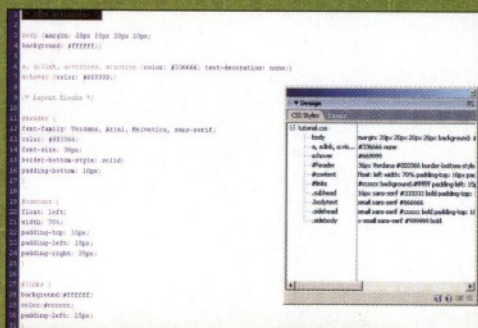
At line one we have a DOCTYPE declaration, defining the document as XHTML 1.0 Transitional, followed in line two by an opening <html> tag with an XML namespace



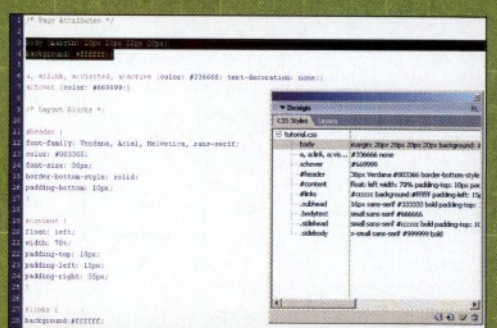
This is followed by a Content-Type declaration, setting the Western European ASCII character set as the page's text type. Dreamweaver automatically inserts these declarations when you create an XHTML compliant document.



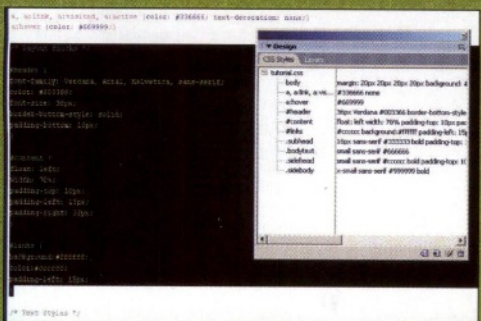
There's an external link to our stylesheet at line six before the head section closes. In the body section, we've commented each block so you can tell which section is which – you'll notice that there's no inline formatting at all.



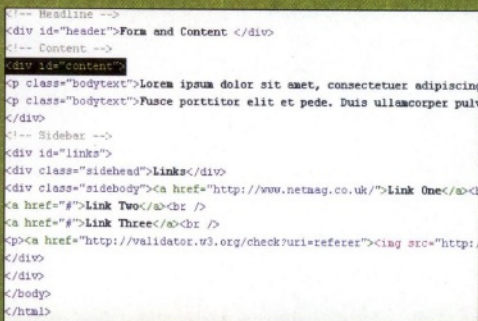
Let's turn to the stylesheet tutorial.css to find where that formatting is. You'll note that the document is commented into sections – this isn't necessary for standards compliance but is good design practice.



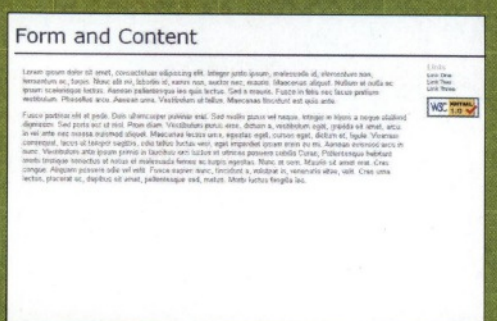
The first section, Page Attributes, defines formatting in fixed elements within our document, in this case the body of the page and hyperlinks. These definitions were defined inline in previous standard versions of HTML.



The next section, Layout Blocks, defines the layout of our page, using CSS ID selectors (ID selectors can only be used once in each document). There are three distinct blocks: Header, Content and Links.



If you flip back to the HTML document for a moment, you'll notice that each commented block is encapsulated between <div> tags declaring these IDs are creating a CSS layout box around each section.



The final section in our CSS file, Text Styles formats text using CSS Class selectors. You'll find that these are used in <p> and <div> tags to format content in our page. The illustration shows how the finished page looks in a browser.

Building websites for everyone

Use Dreamweaver to make an existing website accessible to every user...

Even though accessibility has become increasingly important over the past few years — and a whole plethora of documents, guidelines and books have been written on the subject — it still seems to be considered by designers as a hindrance to creativity.

Accessibility has long been a forum for coders to argue and debate about best practice coding. Often coders will prove their excellent coding skills by developing accessible sites but giving the design the same-old plain cream background with the same font, and using little imagery. The reverse is also true: designers who create beautiful sites don't always know how to make them accessible to all.

Although there are many different design tips that can help make a really cool-looking accessible site, many accessibility issues can be avoided by adding simple coding elements into the website. For those of us with a flare for design but without a degree in coding, Dreamweaver really comes into its own.

In this tutorial, we take you through the process of updating a website to make it accessible. Rather than starting from fresh, we take an existing site, retro-fitting it using simple Dreamweaver techniques. Don't forget your ultimate goal: to make sure all users can access your site. You should use this as a challenge when coming up with future designs. ➤

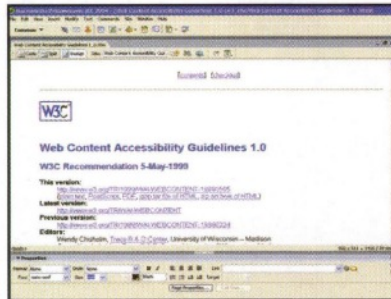


Expertise provided by Dean Russell and Karine Large of iconcertina.creative www.iconcertina.com. For further information about accessibility guidelines visit www.w3.org/WAI and www.macromedia.com



Part 1: Alternative text and your images

Introduce Alt tags as a way of describing images...



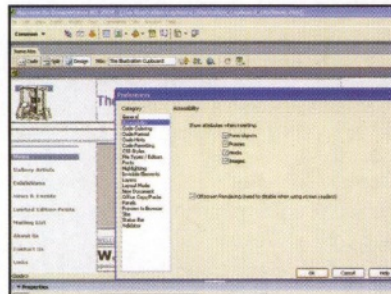
1 First, consider how users may be viewing your site – don't just assume users will have the same PC setup as you. There are many ways in which Dreamweaver can assist you in making sure your site meets best-practice standards – this tutorial covers just some of these. For the full accessibility guidelines, visit www.w3.org/WAI



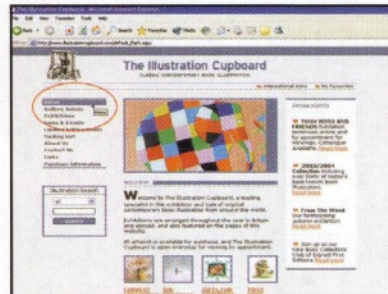
2 Possibly the most important and simplest of all accessibility improvements is to add accurate Alt text to images. This has many benefits, such as helping assistive technologies describe images out loud to users and acting as informative placeholders while images are downloading – due to slow download speeds.



3 Open Dreamweaver. Click Open in the left-hand column of the main screen. Browse and select the page you want to make more accessible. Click Open again.



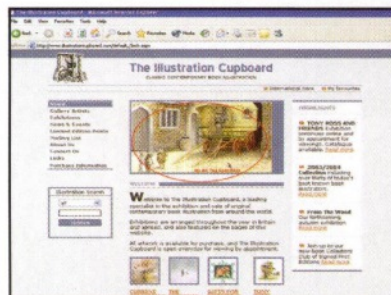
4 Go to Edit > Preferences. Once open, select the Accessibility option. Select the four options: Form objects, Frames, Media and Images. Click OK. By activating these, you'll be prompted to provide accessibility-related information as each element is inserted into a page.



5 Alt text needs to be added to all images, but different types of Alt text descriptions should be used depending upon the image. First, let's look at the navigation. Ideally, this would be text, but if you've used images, you must add Alt text to repeat the same wording included in the navigation.



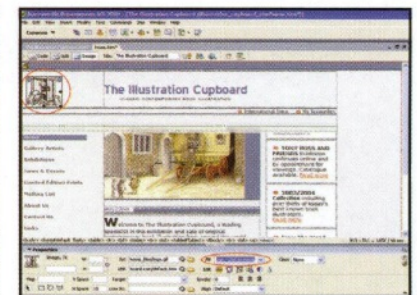
6 Select the navigation image, in this case the Home link, and check out the Properties panel. You'll see a field called Alt. Click in the field and type Home. Repeat this for all your navigation images – adding the same text for the Alt tag as the navigation text. You can choose to add the word 'link' too, but most assistive technologies will tell the user if an image is a link – so it isn't crucial.



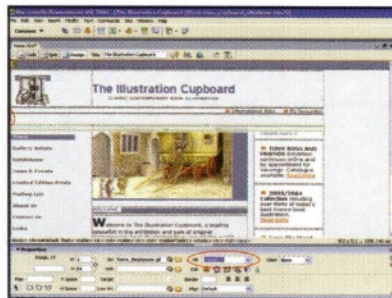
7 Now look at the decorative images on your web page. This is where you need to really think how you want these to be described. For example, the big picture in the middle is an animated gif. This rotates showing different artwork, so your description won't always be accurate. The trick to writing Alt tags is to be descriptive but brief.



8 Select the image, and check out the Properties panel again. Add the text 'Rotating gif of a selection of illustrations' into the Alt field. Then go through all your decorative images and add Alt text.



9 Also, add Alt text to image links. For example, in this page, the top corner logo is a link to the home page (something that every accessible website should have) – so the Alt text here describes the action and the image. We've described it as: Illustration Cupboard Logo – Link to Home.



10 Tiny spacer images must also be updated. Select a spacer gif, then in the Alt field in the Properties panel add <empty> or an asterisk *. This tells the user that the image serves no real purpose.



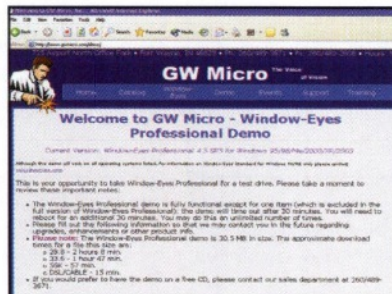
11 After adding all the Alt text to your site, go to File > Preview in Browser and select your preferred browser.



12 When you preview your site in a browser, this new Alt text will appear when you roll-over the image (depending upon the browser you use).

Part 2: Tables and CSS

Find out how to create tables and CSS properly...



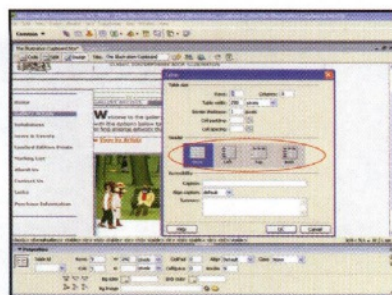
1 Before creating accessible tables, it's important to understand how screen readers interact with tables. You may want to download a free version of a screen reader (window eye) to fully understand how they work. You can find this at [w] www.gwmicro.com/demo.



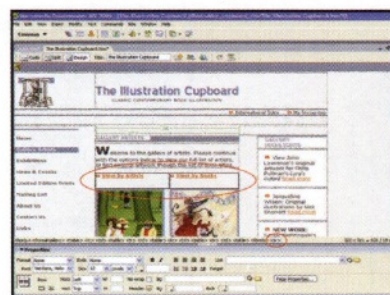
2 Tables are generally used in one of two ways: to control the layout of a page or to present data. For data tables, users read the top row from left to right, and continue reading across the following rows. This requires the user to remember the column/row names and positions.



3 It's simple to solve this issue in Dreamweaver. All you need to do is add column and row headers. If you're adding a new table, go to Insert > Table, where you'll be presented with a dialogue box with a set of choices for the table layout and so on.



4 The important part for us is the Header section, where you'll see four different layout images. None means you don't want any headers; Left is for row headers only; Top is for column headers only and Both is for row and column headers.



5 We want to amend an existing table, so cancel the previous dialogue box. You'll see that column headers are required but not row headers. To make this easier, just select the entire top row, which contains all the cells you want to become column headers (View by Artists, View by Books).

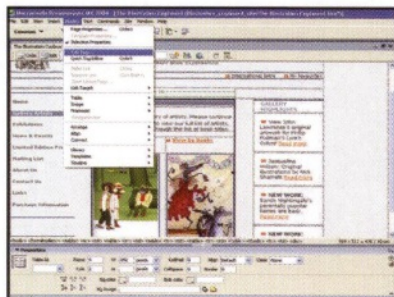


6 Now examine the Properties panel. In here you'll see a checkbox called Header – select this checkbox. These cells are updated to become row headers.

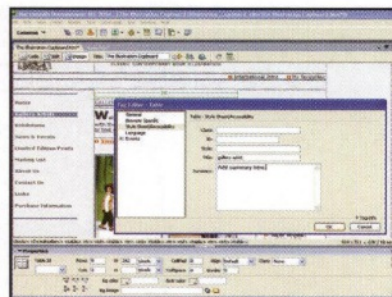


Part 2: Tables and CSS

Continued...



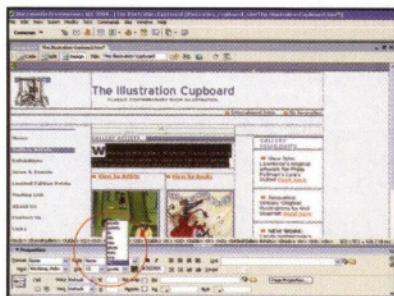
7 When adding a new table, Dreamweaver will prompt you to add other information. In this case, the table is already created so you may want to edit these now. To edit the table information, select the whole table and go to **Modify > Edit Tag**.



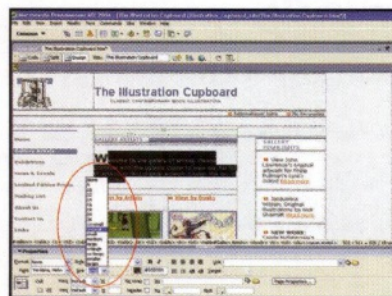
8 In the Tag Editor dialogue box, select **Style Sheet/Accessibility** and fill in the **Title** and **Summary** fields. A table summary is useful for assistive technology users, because it can describe the purpose of the table quickly. Click **OK**.



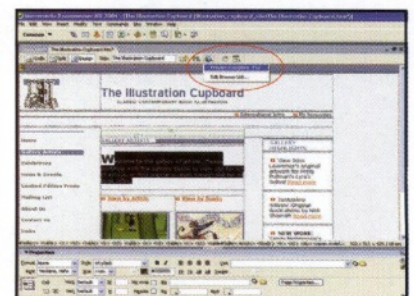
9 A major issue with many users is the size of the text on screen. Designers like tiny text, but it's possible to make the text resizable by the user. Always make sure the text sizing in your site is not in pixel but in em or % so that the user can resize the text via their browser.



10 Highlight the text and look at the Properties dialog box. Change **Pixels** to **%** or **ems**.



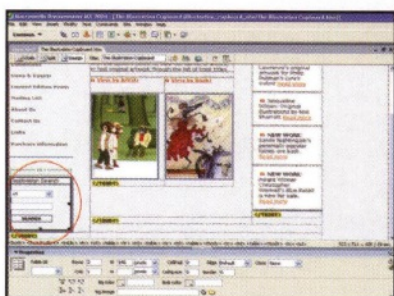
11 Pick the appropriate font size so that the text remains visually the same. Advanced users may want to open the style sheet file and make changes directly from there.



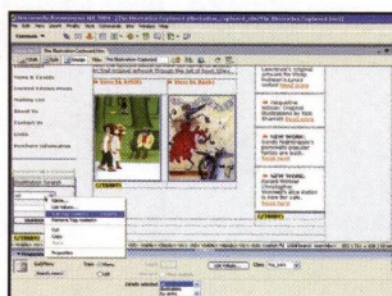
12 Preview in browser to check out your font size on screen. Go to **View > Font Size** and change the selection – for example, to **Large**. As you'll see, the text on screen will increase so it's easier to read.

Part 3: Forms and extra features

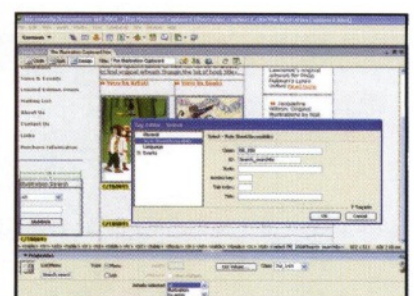
Set up keyboard shortcuts so users can move around quickly and easily...



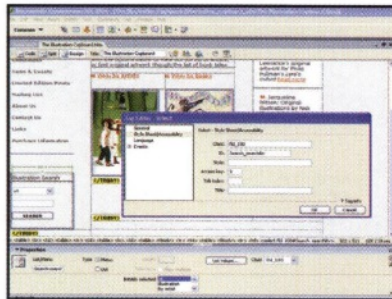
1 You can add access keys or tab indexing to your site to enable all users to navigate through your site easily, and also to make your site especially helpful for users with mobility impairments such as **RSI**.



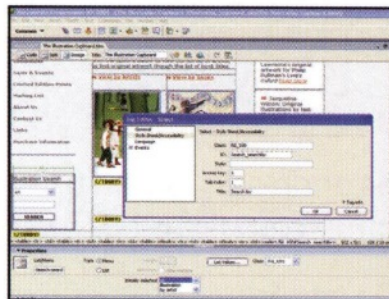
2 Within this site there's a simple search form. Although we aren't going to change the coding for this to work, we're going to update the HTML elements in Dreamweaver to make it accessible. First, right click (Windows) or **Ctrl-click** (Mac) the Search dropdown field. Select **Edit Tag**.



3 Within the Tag Editor dialogue box, select the option **Style Sheet/Accessibility**. There are several options in here for you to complete, but for now concentrate on the **Access Key** and **Tab Index** fields.



4 Type a letter, such as S, into the Access Key field. This means that, in this example, if a user presses Alt+S, the cursor will move directly to the dropdown field. Keep the dialogue box open for the next step, where you'll add a tab index order for the field to specify the order in which the Tab key will move among the individual page elements.



5 The default tab order is determined by the order in which the web page objects appear in the HTML code. If this order differs from the visual layout, a tab index may prove helpful. Type 1 into the Tab Index field. This means the user will be able to tab to the search facility first when entering the page. Repeat the same action for the search form text field, using different tab order numbers and access keys.



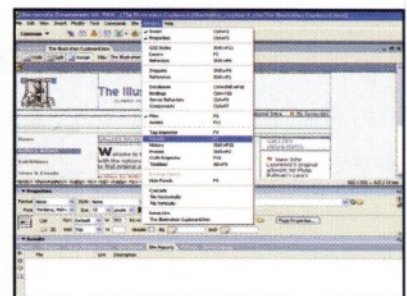
6 Although tab ordering and access keys are helpful, it's often a laborious task for users reliant on keyboard-only access to tab through the navigation or repeated text each time they reach a page. Therefore, it's good practice to add hidden anchor links to important elements at the top of a page. Here, we'll add an anchor link taking the user directly to the main content area.



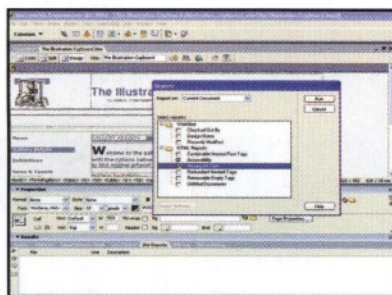
7 Place your cursor somewhere at the top of the page before or after the logo. Go to Insert>Named Anchor.



8 In the Named Anchor dialogue box, type 'Link to content' so that's what the screen reader will read out, giving the user the option not to hear the navigation each time they visit a new page. Click OK.



9 There may be a lot of accessibility issues which appear throughout your site but aren't easy to find. Dreamweaver has a built-in Accessibility Validation tool, which will not only tell you about the issues but show you where they are! To use this tool, go to Window > Results.



10 In the Results dialog box, click on Site reports. Clicking on the green triangle will open the Reports dialog box. Select the reports criteria you want to see (for example, Alt tags and Accessibility issues) and click run. Dreamweaver creates a report, which points out all the elements that are not accessible. Go through them one by one and make amendments as necessary.



11 A sitemap can help users to find key areas of your site that may be in sub-navigation areas, and it can be crucial for users with screen reader to see and navigate the whole site without having to tab through a page with difficult layout. We'd also recommend repeating the main navigation at the bottom of every page so users don't have to scroll back to the top when they reach the bottom – just imagine how frustrating it is to tab all the way back to the top just to click on another navigation link!



12 A final important fact to look at is all the 'read more' links. When using tabs to go through the site, the tabs will jump from one link to another and read them out loud. If the link only says 'read more', it won't mean much to the user hearing it. Instead, use contextual links such as 'read more about accessibility' so the 'read more' link is specific to each story. In this case, we have deleted the 'read more' link and made the text a link. ■



Accessibility and compliance



All the components needed to complete this tutorial can be found on the cover disc.

Tom Knowles addresses accessibility this issue and creates a fully standards-compliant HTML template

Standards this. Accessibility that. XHTML, CSS. That's all we hear these days. Increased accessibility for visually impaired users and multiple devices sounds great. What are these standards, though, and just how realistic is total compatibility?

Current W3C standards call for adherence to XHTML and CSS guidelines, and for the jettisoning of non-structural HTML that's

used for presentation rather than the structuring of documents.

These days, nearly all of the modern browsers fully support CSS2 and the XHTML markup standard, apart from Internet Explorer, which, unfortunately, is also the most widely used browser on the planet.

The overall goal of this standards compliance is to — as much as it is possible — separate the content from its presentation,

and create leaner, more flexible and more compatible markup.

Here, we'll design a web layout in Fireworks and use Dreamweaver to create the site in pure, standards compliant XHTML and CSS.

Part 1: Designing your site template

You should always have a design template before you write any code...

Push boundaries

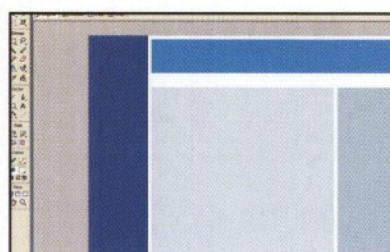
Beware of combining relative and absolute positioning. Poor adherence to standards, means IE struggles to cope if you try anything adventurous. If you insist on pushing boundaries, you might have to spend a while on workarounds and fixes.



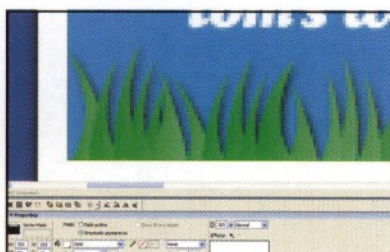
1 First, open Fireworks MX 2004 and create a new document with dimensions of 1024x768 pixels. No matter what the project is, there should always be a design stage before any code is written. Doing this enables you to see exactly how the end result will look, and enable you to get the design absolutely right.



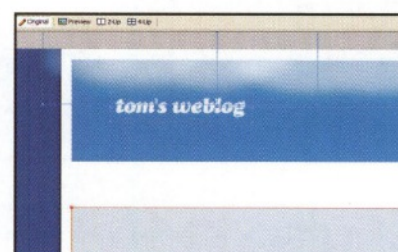
4 You don't have to be a seasoned illustrator to create elements such as this grass — it's just a case of being patient and methodical. Draw a circle and delete one of its points using the subselection tool. Then warp it into a blade shape, before filling it with a green gradient and adding a subtle drop shadow.



2 In the document properties panel, choose a suitable background colour. As you can see from the screen shot, we've arranged our basic layout into several boxes. As ever, it's important to consider the technology you're working with at this stage. CSS is well suited to panel-based designs.



5 Once you've arranged your header area, group all the elements and overlay a white triangle with dimensions that match the original header rectangle. Choose Modify > Mask > Group As Mask. Note that you can vary how much the underlying content shows through by altering the colour of the mask.

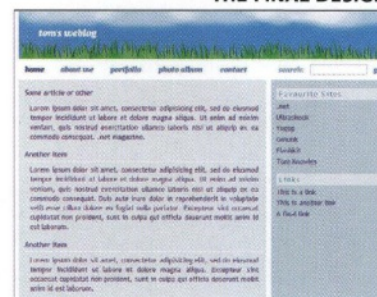


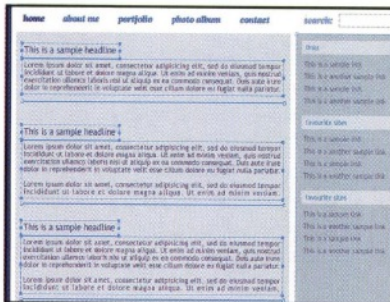
3 We're going to use Cooper Black Italic for all of our site graphics. Choice of typeface is very important in any area of design — once you've settled on a font, stick with it. To create the clouds in the header, use the oval tool, overlaying various transparencies. For a final touch, make use of Gaussian Blue (in the Filters menu).



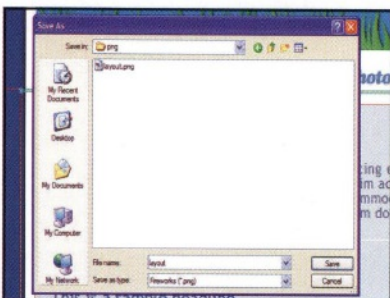
6 For the purposes of this tutorial, we're not going to build an active navigation bar, but we'll still design one at this stage. It's always important for any navigation to show the user which section they are currently in. We'll show this by altering the transparencies (current section is 100%, the others are 80%).

THE FINAL DESIGN

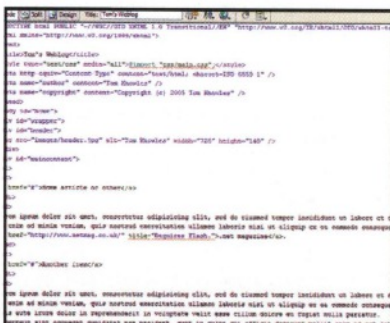




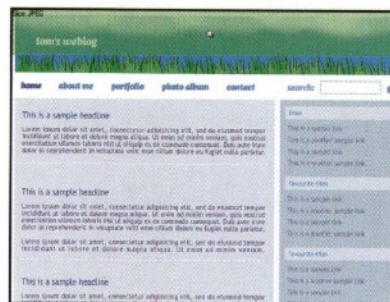
7 Lay your body and supplementary copy out as above. Don't get carried away with typefaces at this stage: CSS can accomplish some neat tricks and some excellent typographical control, but remember that we're limited by the fonts installed on the user's machine. We've used Trebuchet MS – a fairly common font.



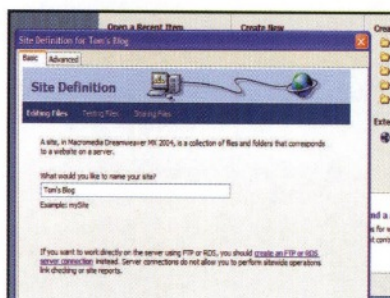
10 With our Fireworks tasks complete, all you need to do is save your file. Fireworks' native file format is an extended version of PNG. This is an incredibly useful format, since it remains editable (including layers, vectors and swatch colours). It's also almost universally supported and hence viewable on most platforms.



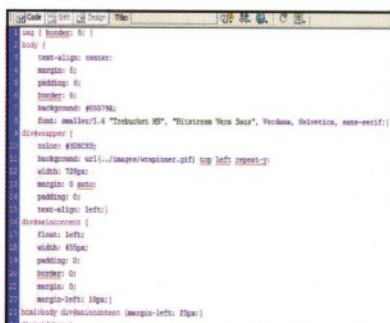
13 Choose Create New > HTML, and type in the HTML in part one of our code. You can preview your page at any time by clicking 'Preview/Debug in Browser' (the little globe icon), although since we haven't written our CSS file yet, your page will not contain any formatting. Save your page in your root folder as index.html.



8 With our design almost complete, use the slice tool to highlight the header bar. Ensure that the slice includes any shadow detail you added to the background, and note down the width for future reference. Choose Export, and select Export Selected Slices Only and save your header graphic as a GIF or JPEG.



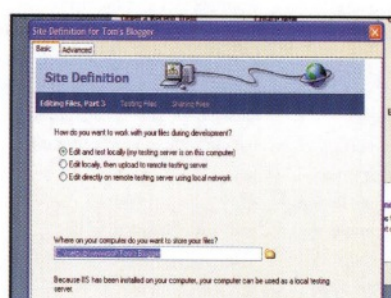
11 Open Dreamweaver MX 2004 and choose New Dreamweaver Site. Follow the wizard – we don't need any server technologies or FTP connectivity at this stage. Choose the parent location of your PNG files as your base folder, confirmed your details and then click Done.



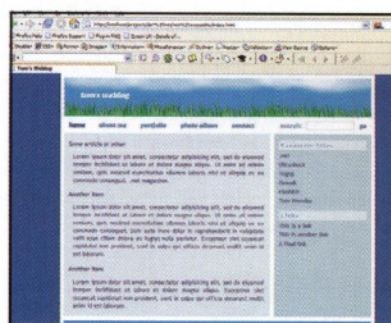
14 Now choose Create New > CSS, and copy the CSS from part two of our code. Dreamweaver provides lots of excellent features for writing CSS, including a sharp code hinting system that removes the need for constant checking of reference books. It also features clear colour formatting to make your code more readable.



9 Most modern sites have a footer, an area at the bottom of the layout that contains copyright information and the like. Once you've finished yours, slice and export it as before. Finally, create a two pixel high slice as above, making sure it is the same width as the header and footer, and export it as wrapinner.gif.



12 A great deal of web design purists still avoid Dreamweaver because of its tendency to generate bloated code. This might be true, but there's more to the package than just WYSIWYG editing. If you hand code all your HTML/CSS, Dreamweaver's organisational facilities are incredibly useful.



15 That's all there is to it! You now have a fully standards-compliant, accessible layout. If you've got Firefox's Web Developer toolbar, try turning styles and images off – you'll see how the page retains a linear flow rather than becoming a horrible mish-mash of nested tables. The file size will also be a great deal smaller.

Know your limits

Understand the limitations of CSS. If you're a tables veteran, you'll be surprised at how limited CSS can initially seem. Consistent styling and total compatibility are factors, so don't be afraid of implementing a hybrid solution.



Part 2: The clean, simple structural XHTML markup

XHTML and CSS layouts are not always as compatible as you might think...

At the risk of irritating a large number of standards purists, the current mish-mash of browsers (the main culprit being IE) means that XHTML and CSS layouts are not as compatible as they would have us all believe. Super-lightweight, compatible markup might be possible on reasonably simple layouts (the two column layout we're building that you see almost everywhere these days, for example), but try anything a bit more complex and you'll see just how far CSS has to go to beat the consistency and dependability of table layouts.

The most sensible solution is to use hybrid layouts. By best of both CSS and tables, you can still write neat, tidy XHTML and have your sites validate properly. Returning to the concept of pure XHTML and CSS, the code below demonstrates just how lean your HTML can be if you're after a simple layout. In cases like this, using tables would actually make your job more laborious.

1 Lines 1–3: The document type declaration is extremely important. This is because it tells the browser what type of document it's looking at, and hence what sort of rules to apply when rendering the content on screen. Its effect can vary depending on how well written your code is; if it's well-formed and sticks to the rules, you might not notice much difference between an HTML 4 document and an XHTML1.1 document. If your code uses a bunch of sloppy, IE-only tricks, prepare for a rough ride if you've declared that your code is valid XHTML.

```
<!DOCTYPE html PUBLIC
"-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/
DTD/xhtml1-transitional.dtd">
<html xmlns="http://
www.w3.org/1999/xhtml">
```

2 Lines 4–10: There are several methods of getting an external style sheet into a document – the @import syntax is the most compatible. Notice how we also stick a couple of meta tags in; these deal with file type and author information. As you can see, XHTML requires all elements to be closed. This includes single tags such as <style> and <meta>.

```
<head>
<title>Tom's Weblog</title>
```

```
<style type="text/css"
media="all">@import "css/
main.css";</style>
<meta http-equiv=
"Content-Type" content="text/
html; charset=ISO-8859-1" />
<meta name="author"
content="Tom Knowles" />
<meta name="copyright"
content="Copyright (c) 2005
Tom Knowles" />
</head>
```

3 Lines 11–15: To achieve table-like grid structures, we need to identify portions of content via the <div> tag. A lot of people are still unsure as to what this ancient tag is supposed to do: put simply, it just divides content up into logical pieces. As part of HTML original purpose, this was a simple way of structuring documents. Thus it makes sense that it is an integral part of standards-compliant markup – by arranging our document into structural <div> tags, the content is readable and usable without any formatting.

```
<body id="home">
<div id="wrapper">
<div id="header">

</div>
```

4 Lines 15–46: Within our wrapper div, we've got a four further divs. These are header, maincontent, sidebar and footer. Since our markup doesn't need to be presentational at all, we can place these divs in an order, based purely on a logical progression. For example, if we were reading the document without any fancy formatting, we would look at the header first, then the main content, then the side bar and finally the footer. If our XHTML were using nested tables, this logical order might well end up spreading the content all all over the place.

```
<div id="maincontent">
<dl>
<dt>
<a href="#">Some article or
other</a>
</dt>
<dd>
Lorem ipsum dolor sit amet,
consectetur adipisicing elit,
```

```
sed do eiusmod tempor
incididunt ut labore et
dolore magna aliqua.
Ut enim ad minim veniam, quis
nostrud exercitation ullamco
laboris nisi ut aliquip ex ea
24. commodo consequat.
<a href="http://www.netmag.
co.uk/" title="Requires
Flash.">.net magazine</a>.
</dd>
<dt>
<a href="#">Another item</a>
</dt>
<dd>
Lorem ipsum dolor sit amet,
consectetur adipisicing elit,
sed do eiusmod tempor
incididunt ut labore et
dolore magna aliqua.
Ut enim ad minim veniam, quis
nostrud exercitation ullamco
laboris nisi ut aliquip ex ea
33. commodo consequat.
Duis aute irure dolor in
reprehenderit in voluptate
velit esse cillum dolore eu
fugiat 35. nulla pariatur.
Excepetur sint occaecat
cupidatat non proident, sunt
in culpa qui officia deserunt
mollit anim id est laborum.
</dd>
<dt>
<a href="#">Another item</a>
</dt>
<dd>
Lorem ipsum dolor sit amet,
consectetur adipisicing elit,
sed do eiusmod tempor
incididunt ut labore et
dolore magna aliqua.
Excepetur sint occaecat
cupidatat non proident, sunt
in culpa qui officia deserunt
mollit anim id est laborum.
</dd>
</dl>
</div>
```

5 Lines 47–62: Our markup also makes use of the definition list tags: <dl>, <dt> and <dd>. Definition lists vary only slightly from other types of lists in that list items consist of two parts: a term and a description. They're ideal for title and subcontent sections such as our main content and the

side bar. Notice that each link has a title attribute to explain its purpose.

```
<div id="sidebar">
<dl>
<dt>Favourite Sites</dt>
<dd><a href="http://
www.netmag.co.uk" title="
net">.net</a></dd>
<dd><a href="http://
www.ultrashock.com" title="
Ultrashock">Ultrashock
</a></dd>
<dd><a href="http://
www.yugop.com" title="Yugo
Nakamura">Yugop</a></dd>
<dd><a href="http://
www.gmunk.com"
title="Gmunk">Gmunk</a></dd>
<dd><a href="http://
www.flashkit.com" title="
Flashkit">Flashkit</a></dd>
<dd><a href="http://
www.tkl.co.uk" title="
Tom Knowles">Tom Knowles</
a></dd></dl>
<dt>Links</dt>
<dd><a href="#" title="">This
is a link</a></dd>
<dd><a href="#" title="">This
is another link</a></dd>
<dd><a href="#" title="">A
final link</a></dd>
</dl>
</div>
```

6 Lines 63–68: Finally, we mark up our footer image. Since we're using CSS for positioning, there's really no need to specify the image dimensions, all it does is bloat the amount of code. One attribute the tag does require is alt. This is an accessibility requirement; if the user has chosen not to download images (or their browser cannot view them), they need to be provided with an indication of what the image is.

```
<div id="footer">

</div>
</div>
</body>
</html>
```




Part 3: Performing presentational magic with the CSS file

The trick is to ignore the difference in syntax...

For HTML coders reared on a diet of Notepad, super-nested tables and spacer gifs, CSS's relative and absolute positioning system can seem a bit random and messy. The trick is to ignore the difference in syntax and approach the code in exactly the same way you always have done: formulate your overall structure on paper, and start with a simple framework. Once you've got your basic CSS layout working efficiently, you can begin to concentrate on individual areas. The following code first deals with presentational aspects such as links and text formatting. It then moves on to the page geography and, finally, individually positioned elements. For better organisation, it's always a good idea to break this down further – a completely separate stylesheet just for text formatting, for example. By taking this approach it becomes simple to make quick changes, without wading through some of the more complex CSS that may be required.

1 Lines 1–8: If you've ever tried to hyperlink an image tag (without specifying a `border=0` attribute), you'll know that HTML places an ugly border around the image. Rather than bloating our markup with loads of these border attributes, we can simply use CSS to ensure that the borders are no more. In this code chunk, we also take control of the body element, specifying the padding, margins and border as well as the background colour and typeface. Notice how we provide a number of fonts ranging from the ideal (Trebuchet MS), to as-close-as-possible.

```
img { border: 0; }
body {
  text-align: center;
  margin: 0;
  padding: 0;
  border: 0;
  background: #05579A;
  font: smaller/1.4
    "Trebuchet MS", "Bitstream
    Vera Sans", Verdana,
    Helvetica, sans-serif; }
```

2 Lines 9–15: Our wrapper div. To centre it in the browser, we set a width (in this case 728 pixels), and then give it an auto margin setting. Note the background graphic. This is the GIF we created earlier on. XHTML doesn't support 100 per cent heights as such, so to mimic 100 per cent table cells, we must resort to cheating and using a repeating graphic background. This isn't a major problem, though, it's still easy to update and is fully compatible.

```
div#wrapper {
  color: #3D5C8D;
  background: url(../images/
    wrapper.gif) top left
    repeat-y;
  width: 728px;
  margin: 0 auto;
  padding: 0;
  text-align: left; }
```

3 Lines 16–33: This is where we set up our three main divs. Anyone who tells you that CSS isn't pixel-precise is talking nonsense. Here we set pixel-based alignment and sizing. Alas, this is also the place where we run into browser inconsistencies. Internet Explorer doesn't use the CSS box model properly, so we have to use `html>body` syntax (which IE can't parse) to reset the margin after we've first

declared it.

```
div#maincontent {
  float: left;
  width: 455px;
  padding: 0;
  border: 0;
  margin: 0;
  margin-left: 10px; }
html>body div#maincontent
{ margin-left: 25px; }
div#sidebar {
  margin: 0 0 0 504px;
  padding: 0;
  width: 200px;
  border: 0; }
div#footer {
  border: 0;
  width: 728px;
  padding: 0;
  margin: 0px 0 -4px 0; }
```

4 Lines 34–41: Here we're getting into `h1` links. The `display: block` part means that each `h1` tag occupies its own horizontal space. Of course, HTML does that with `h1` tags by default, but with CSS we could change that if we wanted to. The anchor sub-tag has several states – a live link, visited, hover etc. Manipulating these states is simple: we just use `a:visited` syntax.

```
h1 a:link, h1 a:visited {
  display: block;
  width: 700px;
  height: 50px;
  margin: 0 0 0 15px;
  padding: 0;
  border: 0; }
```

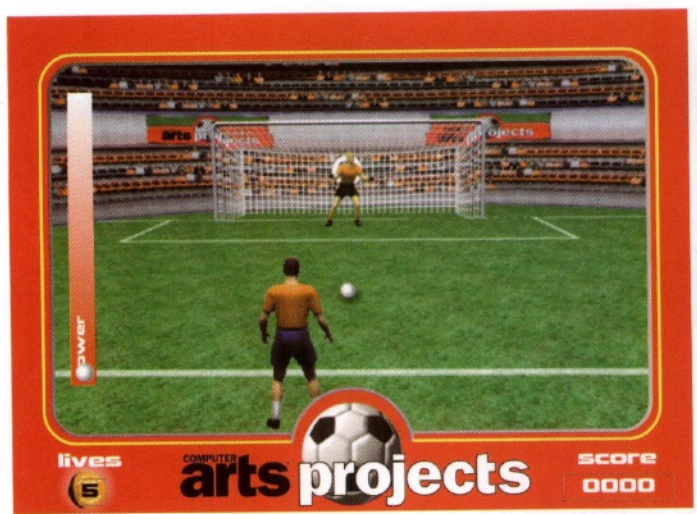
5 Lines 42–64: CSS's typographical control is really where it comes into its own. It might be a little clunky when it comes to complex layouts, but when used properly, we can deliver incredibly detailed and slick typesetting. For example, check out letter-spacing – which does exactly what you'd expect. Another great idea is text-transform – we've not used it here, but it allows you to switch between upper and lowercase. This sort of functionality ventures into the realm of client-side scripting, and is hugely useful.

```
a:link, a:visited {
  font-weight: bold;
  text-decoration: none;
  color: #05579A; }
a:hover {
  font-weight: bold;
  text-decoration: none; }
```

```
color: #0C8DCB; }
div#maincontent p a:link,
div#maincontent p a:visited {
  text-decoration: underline; }
div#maincontent p a:hover {
  text-decoration: none; }
p { margin: 0 0 1em 0;
  line-height: 140%; }
#sidebar dd, #sidebar dt {
  margin-left: 0;
  padding-left: 0;
  line-height: 18px;
  margin-bottom: 3px; }
body #sidebar dt {
  letter-spacing: 2px;
  margin-top: 15px;
  color: #0C8DCB;
  padding: 2px 5px;
  background: #DBE6EB; }
#sidebar dd a:link,
#sidebar dd a:visited {
  font-weight: normal; }
```

6 Lines 65–75: More margin and padding information. We can be very specific about padding and margins – rather than applying the same value to all sides of an element, we can set different values for each. Additionally, CSS's version of type leading is called line-height. The cleanest, most readable setting is usually between 120 and 140 per cent. Naturally, this depends on the typeface and, if you're struggling for space, you're probably not going to want enormous gaps between each line of text.

```
div#maincontent dl {
  margin-top: 15px;
  padding-top: 0; }
div#maincontent dl dt {
  padding-bottom: 5px;
  margin-bottom: 5px;
  border: 0;
  border-bottom: 1px solid
    #ffffff;
  line-height: 140%; }
div#maincontent dl dd {
  margin: 0 10px 15px 10px;
  padding: 0;
  text-align: justify; }
```

Dreamweaver Interface Guide (Desktop/DreamweaverGuide.php*) (XHTML)

Commands Site Window Help

Title: Dreamweaver Interface Guide

Guide</title>

10 margin-top: 0px;

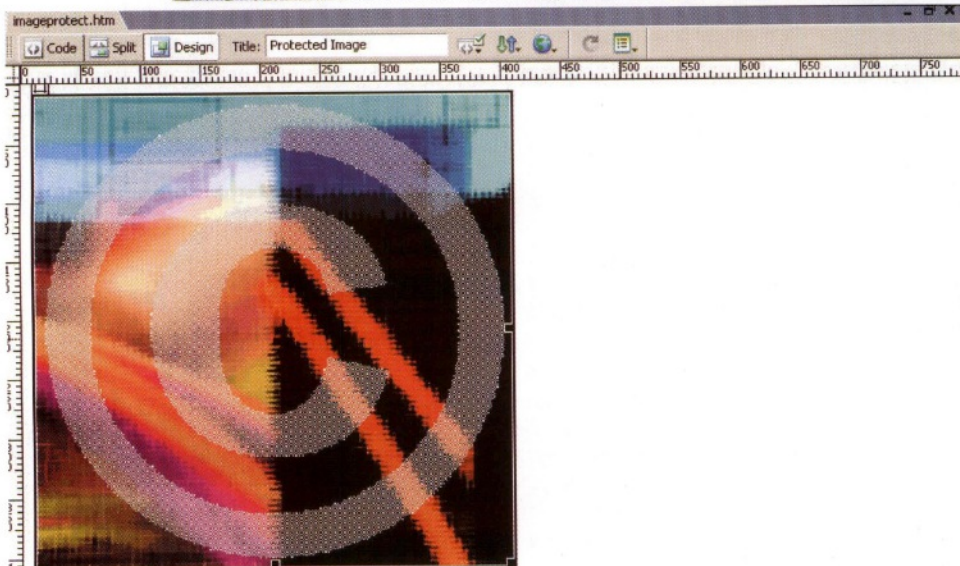
BD4D FEATURE



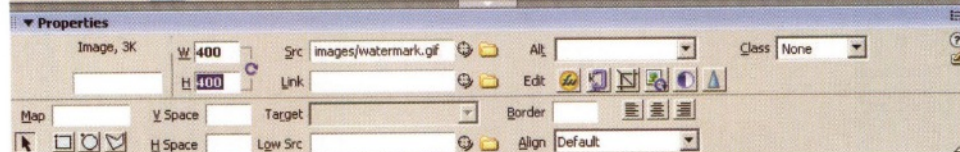
<body>

643 x 159 252K / 70 sec

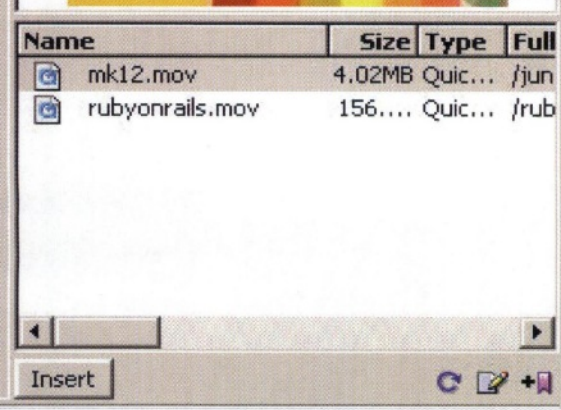
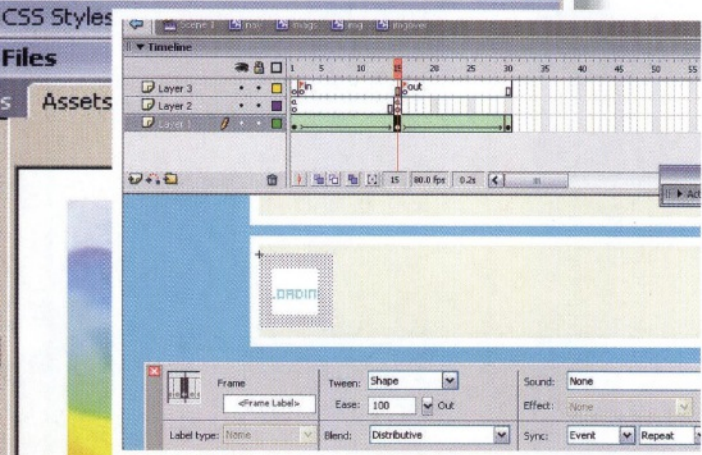
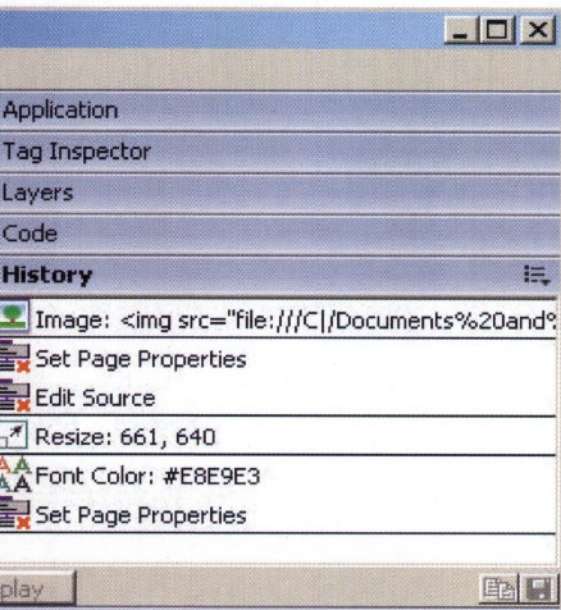
Timelines



<body> <div#Layer1>



2



Studio MX 2004 projects

The beauty of Macromedia's Studio MX suite lies in its consistency. This chapter shows you how Flash interoperates with other elements such as Fireworks and Dreamweaver in a complimentary way...

62	PROFILE: Zoolook
68	PROFILE: Hi-Res
70	Building games with Studio MX 2004
74	Create games in Flash
80	Shoot-'em-ups
84	Understanding the Dreamweaver interface
86	CSS-based layers
90	Virtual Tour
94	Photo Album
98	Flash Guestbook
102	Contribute
106	Protect your work

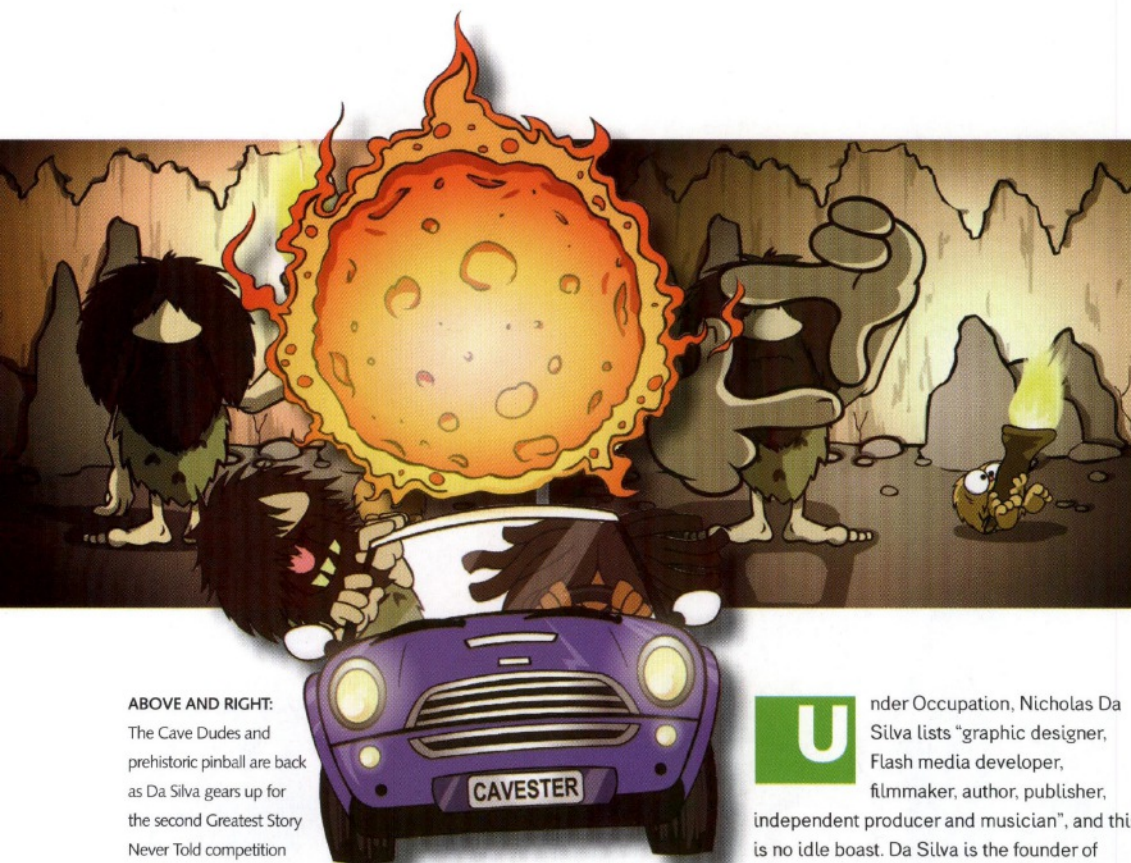




ZOOLOOK®

ON HIS WEBSITE, NICHOLAS DA SILVA IS KNOWN AS THE DREAMER. BUT WITH A LIST OF JOB TITLES AS LONG AS HIS ARM, AND A RELENTLESS STREAM OF TOP CLASS PROJECTS AND CLIENTS HEADING HIS WAY, THERE'S PRECIOUS LITTLE TIME FOR DREAMING. ✕ ✕

WORDS BY ED RICKETTS
PHOTOGRAPHY BY JON HOPE >



ABOVE AND RIGHT:
The Cave Dudes and prehistoric pinball are back as Da Silva gears up for the second Greatest Story Never Told competition



COMPANY INFO

Call Zoolook on:

+1 415 775 9549

Email: info@zoolook.com

or visit www.zoolook.com

FlashTV: www.flashtv.com

The Greatest Story Never Told:

www.tgsnt.com

Dread and Alive: www.dreadandalive.com

Hitless: www.hitless.com

Under Occupation, Nicholas Da Silva lists "graphic designer, Flash media developer, filmmaker, author, publisher, independent producer and musician", and this is no idle boast. Da Silva is the founder of Zoolook, a new media studio that boasts clients such as TechTV, and the brains behind FlashTV, a web and TV showcase for Flash artists that runs in his adopted home of San Francisco.

He's also the curator of The Greatest Story Never Told, an ongoing digital storytelling competition, and writes the Dread & Alive and Hitless comic book series. In his spare time, he composes music, programs 3D games and somehow still finds time to help his wife raise their young daughter.

Such restlessness may be a product of an eclectic background. Having travelled the world with his family, he eventually settled down on the east coast of America and attended a Catholic high school. It was there

Founded in 1996 and named after a Jean Michel Jarre album, Zoolook Studios is Da Silva's bread-and-butter. "I also work for TechTV, a group called Front 242, which I'm the official graphic designer for, and for a lot of big companies," he explains. "I'm doing DVD authoring, Flash web sites and presentations. Over the years, Zoolook has been able to create a business model out of its own stuff, so right now I'm like a kid in a candy store, just having fun."

Complementary hours

Since his wife is also an artist — on the non-profit side — Da Silva tries to work complementary hours in order to keep some semblance of family life. "I usually work until one or two o'clock in the morning, though I don't see it as work. You're just in the zone, you start working on something and from that you go on to the next thing."

Right now, he's gearing up for the second

CASE STUDY 1: DREAD & ALIVE



When Cultural Anthropologist Philip McIntosh is summoned to the marooned village of Accompong, Jamaica, to investigate a series of unexplained deaths, he brings along his wife and seven-year-old son, Drew McIntosh. Once inside the forbidden Cockpit Country, their visit is anything but tranquil as they soon find themselves in the midst of a supernatural battle between good and evil.

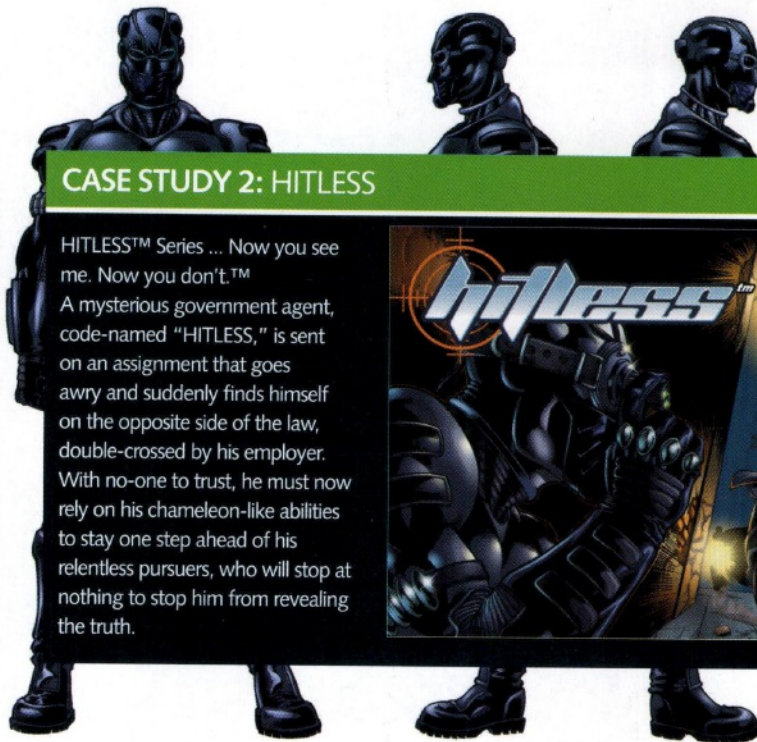
"I usually work until one or two o'clock in the morning, in order to keep some semblance of family life"

that Da Silva discovered his talent for sport. "I had options to do that in college, so I decided to do that and try to be an architect, too," he says. An engineering degree followed.

"I wound up leaving the east coast and coming out to California. I was a skateboarder and a musician, and I met a lot of artists out here. That just blossomed into defining myself as a professional, and it took off from there."

annual instalment of The Greatest Story Never Told (TGSNT). This online competition encourages storytelling in Flash, with awards in a number of categories and entries judged on artistic talent, storytelling and vision. Although last year's event was successful, Da Silva is determined to make this one even better.

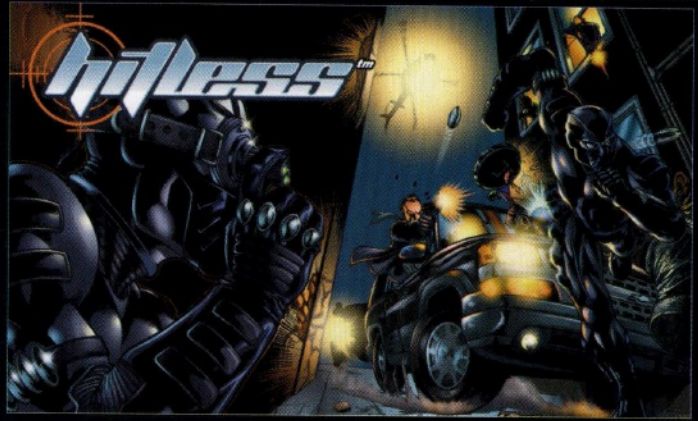
"I was a little nervous at first, because you always are when you create something for the



CASE STUDY 2: HITLESS

HITLESS™ Series ... Now you see me. Now you don't.™

A mysterious government agent, code-named "HITLESS," is sent on an assignment that goes awry and suddenly finds himself on the opposite side of the law, double-crossed by his employer. With no-one to trust, he must now rely on his chameleon-like abilities to stay one step ahead of his relentless pursuers, who will stop at nothing to stop him from revealing the truth.



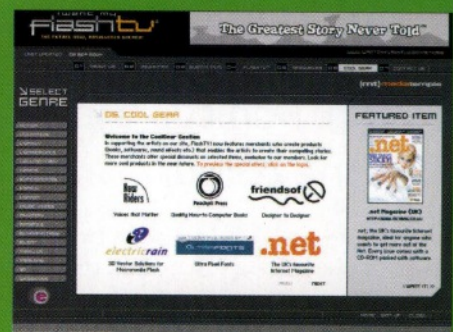
very first time," he says about the first TGSNT. "You try to create something a little bit different and you want to see how people respond to it. The only negative thing about last year's event was that a lot of people didn't find out about it until two weeks before the deadline. So this time we're making sure people are aware of the competition, and it looks as if it'll be bigger than ever."

In tandem, version 2.0 of FlashTV is in development and a November launch is planned, featuring an animated host named Flavia. "I wanted to feature what's going on in the Flash world, not just movies and storytelling," he says. "So games, a bit of wireless work, and so on. I want to make it a site for independent artists that produce any sort of Flash content, somewhere where they can showcase their work."

FlashTV now features more than 500 movies from artists all over the world, although quality is more important than quantity. "We want to be recognised for having really high-quality content from serious artists who have a passion for telling a good story," he says

Cross-purposing

So just what is it about Flash that gets him? "What I really like is the cross-purposing. I like the fact I can create something for the web and then use it elsewhere. I produce a TV series for FlashTV entirely in Flash; we export to DV, and they say it's the best thing on their network [AccessSF in San Francisco]. There's a lot of interest in taking FlashTV to a 24-hour network, which will be great. I want to create a business model where artists can be paid to create content."



LEFT AND ABOVE: FlashTV's Flavia, Nick Da Silva and Serena Guidobaldi host a virtual world of Flash entertainment. All genres are covered, including comedy, adventure, action, horror and sci-fi/fantasy



Da Silva spent seven years researching Jamaican religions before penning his first *Dread & Alive* strip. Myalism and Obeahism became the "good" and "evil" forces battling against each other in his stories

"I'm hoping Macromedia will spend more time on the storytelling for animators, and the sound features [of Flash]. I'm working on a template to create content for HDTV [high definition television]. I think that's critical and I hope that Macromedia will start thinking in those terms, too. You think a couple of years ahead so the artists can do these things now."

"I started on the PC first, on a 286. I used to work in a reggae club and I'd design T-shirts, the poor man's version with just one colour. I wanted to do a four-colour design for the ten-year anniversary of Bob Marley's death, so I called a couple of design places. At the one I chose, a woman was working on a Mac 2si and I watched her take what I did by hand,



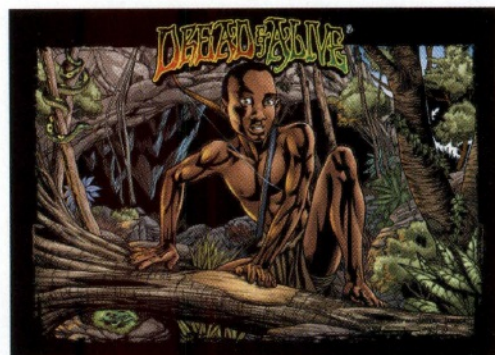
The sinister ShadowCatcher, here seated at his throne inside Windsor Cave, Jamaica, ponders his next move

"I used to work in a reggae club and I'd design T-shirts, the poor man's version with just one colour"

While Flash remains his app of choice, Da Silva has dabbled with, and indeed still uses, a great deal of other software for his creations: ToonBoom, 3ds max, LightWave, combustion, and so on. And unsurprisingly, given that his logo of "Tink Diffra Mon!" is a cheeky homage to Apple, he's a die-hard Mac user with a genuine zeal for the machine.

traditionally, and do it on the computer in two hours. I bought a 2ci, and my first job paid for the computer. I've never turned back. I love Mac," Da Silva admits.

Far from finding his myriad projects distracting, Da Silva prefers to juggle several jobs at once. "I spend half a day, four or five hours, working with clients, and the rest of the time working on Zoolook



Cuffee standing outside Windsor Cave in the Cockpit Country, a rugged and inaccessible area of inland Jamaica

A HISTORY: From there to here, and all points in between...

Da Silva's labour of love – more passion, in fact – has grown so much and so rapidly, he no longer finds he has to pitch for work. "My background is advertising – I worked for an agency for three years," he says. "It was a food service company, so we were working for people like Nestlé and Garden Burger. I started another company in 1999 called Paper Motion, which was a motion graphics company, and I got to know a lot of people in San Francisco. So you'd be handing out your business card and, next thing you know, you're getting calls from someone who has seen your work. That's what usually happens. I try not to turn down any projects, but sometimes you have to. I tend to have three to four client

properties. Sometimes when it comes to creative processes, you get stuck writing a chapter for the comic book, so I think, well, maybe I'll work on some ideas for FlashTV.

"It seems chaotic, but it really helps you when you move on to the next thing. I have a ton of sketchbooks, which I use all the time. Every one of those will get produced – that's how I am. Nothing's wasted."

But not everything is produced solely by Da Silva, of course. He works with Sebastian Gronemeyer, a "very talented" Flash author based in Spain, as well as Floating Pear Productions in Rhode Island.

"With the comic books, I write the story and come up with concepts. I've got a great team in Brazil who do amazing work in breathing life into the characters. It makes things a lot easier when you have a small group of people together who can think alike."

"Be original"

Innovation and independence, it soon becomes apparent, are Da Silva's watchwords. He cites Pollack and the ceiling of the Sistine Chapel as influences: "That's what drives my passion – independence. Ola Bergner from Sweden, Kirk Millett, they produce their visions in Flash, and that inspires me to do the same."

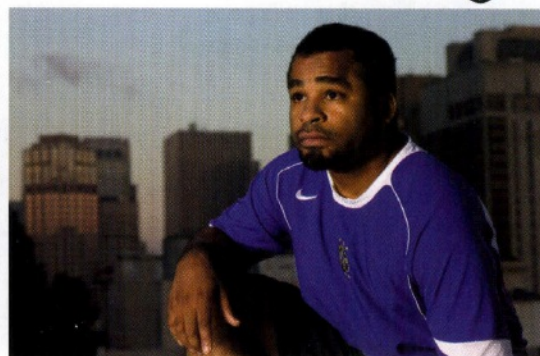
But as the two sides of ZooLook demonstrate, Da Silva's also acutely aware of commercial pressures amidst the creativity. "Right now, everyone is trying to figure out what the next big thing is going to be on the web. During the dot-

com bust, I saw young upstarts with great ideas living a fantasy lifestyle. No-one had business plans, which is the foundation of any business. They had these sugar daddies just keeping the company afloat."

He tells a story of a company that hired the whole of Ellis Island and the Statue of Liberty for a launch back in those dark days. "They're out of business now," he adds somewhat unnecessarily. "If you can't show that you can turn a profit in four to five years, then it's not a successful business. A lot of money was spent wastefully back then; I'm hoping people learned from that mistake."

Anything else upcoming artists should learn? "I would teach them to be original," says Da Silva. "I get a lot of resumés in and they'll include a link to a web site that's just a copy of some popular site. You need to be original, to push the limits of what you can do. I'd also teach them to be independent, and learn the laws of the industry. By that I mean trademarks and copyrights. My father taught me how to copyright my own art when I was just ten years old."

"If you learn to work in that way, you can control your destiny. I love that word: independence."



Name Nicholas Da Silva, a.k.a. the Dreamer

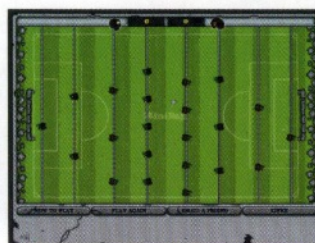
Role Designer, producer, author, animator and musician

From USA/Brasil (currently based in San Francisco, California)

Inspired by My daughter – Mariana Mai "M&Ms" Da Silva, Bob Marley, Jean Michel Jarre, Kitaro, Moby, TCM and BT. Also *Computer Arts Projects* (because, just because), snowboarding at Whistler/Blackcomb and Lake Tahoe and directors Michael Mann, Sam Raimi and Luc Besson for their incredible vision. I love Dean Koontz (the amazing storyteller) and all the artists worldwide who "tink diffran mon!"

Favourite designer/illustrator David Martin, Fantasy Interfaces and Rodney Buchemi of Bigjack Studios

Favourite web site FlashTV.com, FavouriteWebsiteAwards.com and Google.com

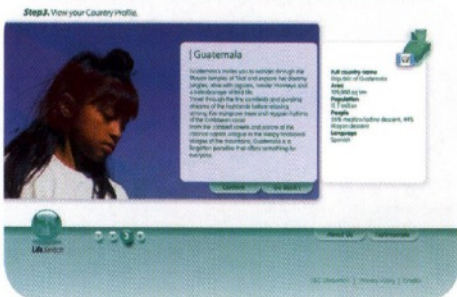
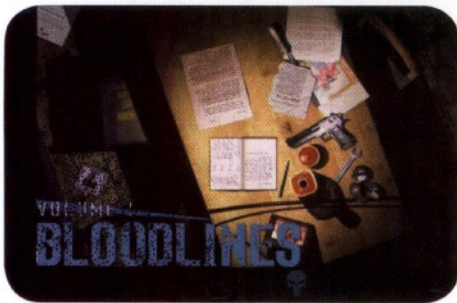


ABOVE AND RIGHT: The Cave Dudes Miniball Game, part of The Greatest Story Never Told web site, was officially selected at the 2004 CINEME International Animation Festival



Hi-ReS!

"You give people the chance to communicate freely and things start to become interesting." Hi-ReS! lifts the lid on what makes interaction work...



Since 1999, Hi-ReS! has been blazing a trail through the world of interactive media, turning heads with a combination of dark humour, visual flourish and a willingness to explore what it means to be interactive.

"Hi-ReS! stands for handsome information, radical entertainment systems," explains Florian Schmitt, one half of the founding duo. "But sometimes we like to call it sponsored weirdness."

This volatile mixture has lured clients such as Lexus, HBO and MTV to the Hi-ReS! experience and led to the creation of groundbreaking sites soulbath.com, thepunisher.com and thethirdplace.com.

The challenge

The rapid switch between playful and thoughtful that characterises its work also helps Hi-ReS! maintain an avowedly challenging edge – "It's our motto," explains Florian. "We want to challenge people."

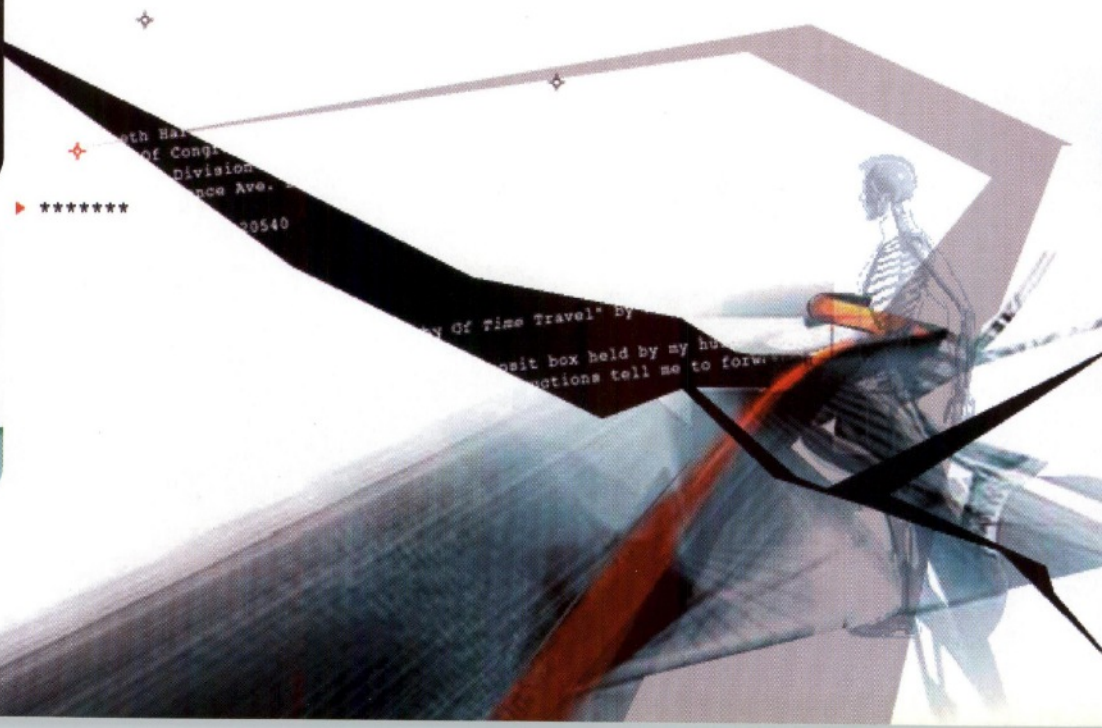
Florian, who judged the interactive category at this year's D&AD Awards, has coined his own phrase for what the company is doing: "I call it assisted discovery.

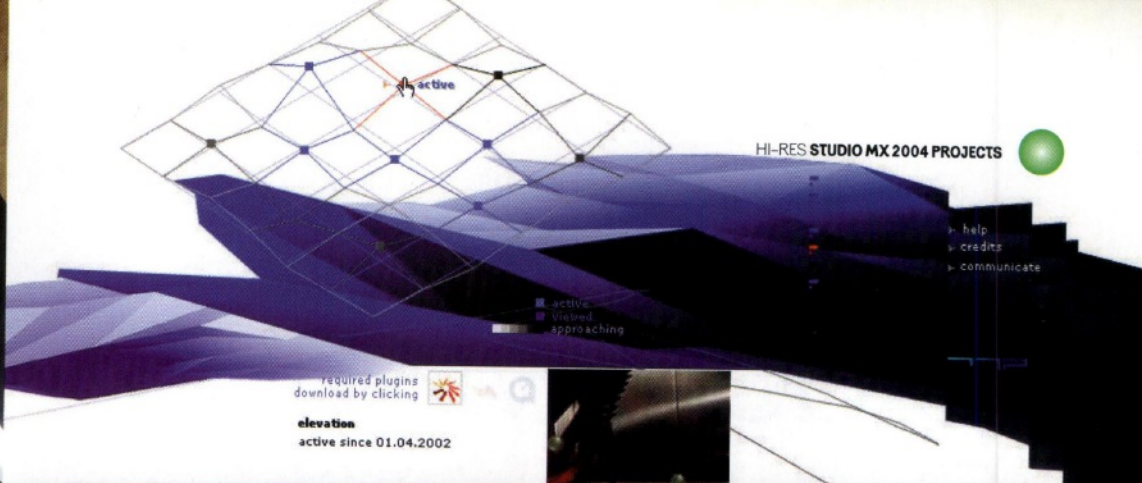
It's about creating understanding, changing something through your interactions."

While he's clearly having a bit of fun, this does give some insight into what makes Hi-ReS! special, why its work is more than equal to the task of promoting films such as *Requiem for a Dream*, *Donnie Darko* and *The Punisher*. "It's about giving people the ability to influence the system," says Florian, as opposed to making the system obvious and therefore boring.

When they set up Hi-ReS!, Florian Schmitt and Alexandra Jugovic had one major ambition: "I think for us, the foremost goal was always to see work as play," says Florian. Unfortunately, there wasn't a barrage of eager punters looking for challenging multimedia content, because by then the internet bubble was already starting to take on a mythical aspect.

So Alexandra and Florian decided to become their own first client, commissioning themselves to produce soulbath.com. "The idea to become our own clients was born out of the fact that no one was really waiting to give jobs to two Germans who had just arrived in London and hadn't done any interactive





work." The way Florian tells it you almost feel sorry for him, but the next chapter of the story fixes that.

The pair sent themselves a brief and, with a deadline set by the rapidly approaching expiry date on their trial versions of Flash, they set to work. "It was more of a fun project to keep us busy," muses Florian.

Celebrating malfunction

Soulbath is lo-fi meets hi-tech in a head-on crash, Florian explains by saying: "We wanted to do something that celebrated malfunction, that's what we were interested in at the time." Along with its associated exhibition of subversive banner ads, ClickHere!, the site won Hi-ReSi some well-deserved attention from, among others, The New York Times.

Hi-ReSi consistently produces sites that feel properly interactive, like they extend beyond your feeble clicking. "It's always difficult to know when to stop," complains Florian. "There's nobody here telling us to go home so we just keep going."

"For us, communication is the creation of understanding (or misunderstanding) through

interaction between two entities." In developing a way to relate a story, Hi-ReSi is "always trying to anticipate this interaction in order to shape our narratives and thus the flow of the web sites." The outcome for each user is hard to predict, but that's how it should be.

Hi-ReSi continues to explore the nature of interaction with a combination of installation work and technological examination. Florian isn't entirely joking when he says, "We're researching 'alternative interactions' with 'the machine' at the moment."

At the Hi-ReSi studio in Shoreditch, work is currently under way on an installation and web site which will support a Barbican exhibition celebrating 40 years of graphic design. "We're working on a gesture-based system at the moment," says Florian, a direction heavily influenced by the fact that, "It's amazing, if you give people the opportunity to draw a penis, they will!"

But Florian's happy with this. It's examining how people choose to communicate, taking their work in unexpected directions. "It's just one of those things. Because it's interactive, you can't predict what will happen every step along the way." ■

OPPOSITE PAGE, FROM TOP TO BOTTOM:

Hi-ReSi delivered titles, set, sound and logo design for MTV: New. The concept used was 'twisted wonders', described as "somewhere between fascination and being unsure."

Following the MTV:New project, Hi-ReSi is soon to unveil its work on an international style guide for MTV. Produced in tandem with fellow Shoreditch studio Precursor, the project is intended to give disparate MTV studios a clear idea of the station's flavour.

The Wake forms one part of the web site Hi-ReSi produced for HBO's Six Feet Under series. This online experience is billed as a metaphysical pitstop, perhaps taking the Hi-ReSi 'sponsored weirdness' philosophy to its logical conclusion.

"The great thing about The Punisher is that he's a super hero, but he doesn't have any super powers - he's just really pissed off," explains Florian.

This site, for the charity Christian Aid, struck gold at the prestigious Cannes Lion Advertising festival this year. The site's glossy self-help parody is so good it has a genuine effect.

"We can do light and happy," insists Florian Schmitt. "It's just that the dark comes more naturally." The site that Hi-ReSi built for Richard Kelly's brilliant debut film, Donnie Darko, is proof that the second half of this comment is truer.

THIS PAGE:

Top left: With the Shiny Vistas installation projects, shown at this year's FEED Festival, scripts and interactions, sound and light are triggered by the viewer's presence.

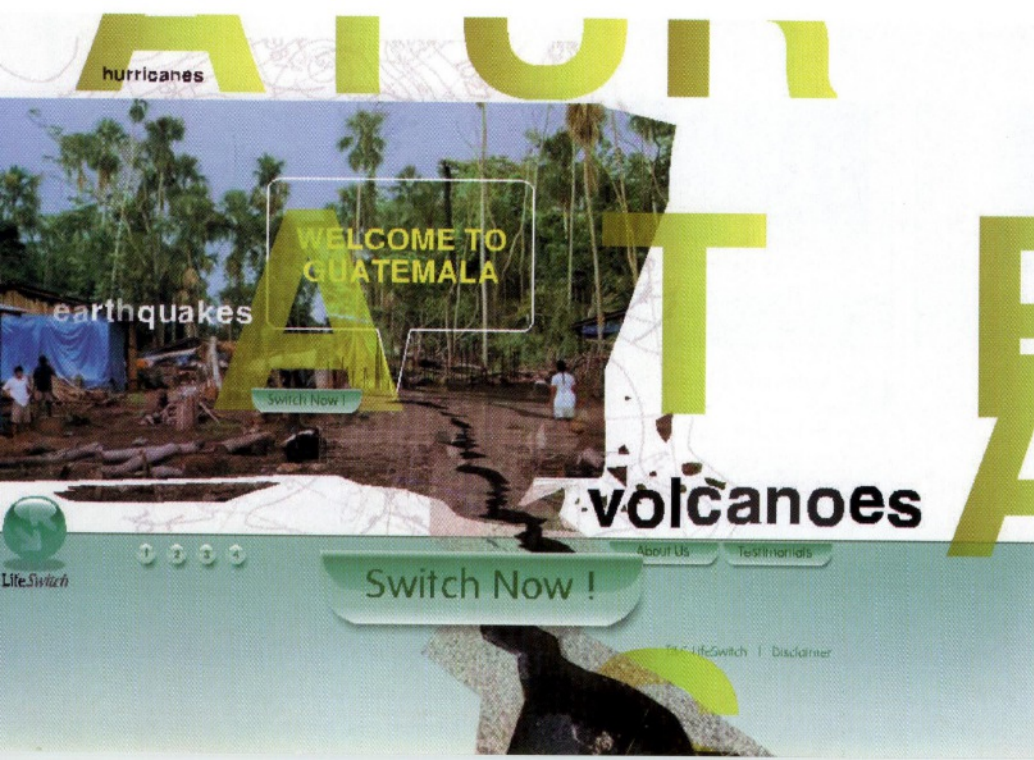
Top right: The multi award-winning site for Sony's PlayStation concept saw Hi-ReSi reprise a curatorial role first seen on soulbath.com, collecting together the impressions of designers and artists from around the world to help define the concept.

Above left: "We're in the enviable situation of being able to turn down work we don't feel suits us," says Florian. Hi-ReSi works with clients, such as Diesel Style Lab, who are willing to give its 'challenging' philosophy a fair chance of success.

Above right: Pioneers in their own right, Massive Attack called on Hi-ReSi to produce these clips from footage of Nick Knight's cover shoot for 100th Window. "It's about giving people the ability to influence the system."

Left: Hi-ReSi succeeds in making interactive experiences worthwhile because it intentionally avoids being obvious about what's on offer. LifeSwitch is a web site for Christian Aid.

You can find out more about Hi-ReSi by visiting www.hi-res.net





Studio MX: Building games



All the components needed to complete this tutorial can be found on the cover disc.

Tom Knowles uses Macromedia Studio MX to design and build a top-down car game

Flash's visual development paradigm and powerful ActionScript scripting language make it ideal for creating web-based games. This used to be the preserve of Shockwave, but Flash is catching up to its big brother. A few years ago, you could tell a Flash game from Shockwave game by the

quality of the graphics. Shockwave's powerful engine was suited to pushing BMPs around, hence any multimedia created with it usually looked richly graphical and sumptuous, while Flash struggled with BMPs. The slow processors of the day meant most games were rather cheesy all-vector affairs.

These days, Shockwave tends to be the

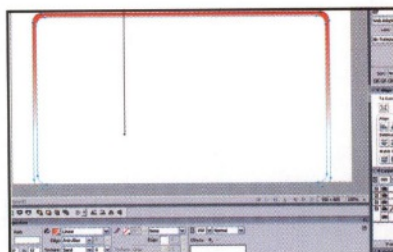
platform of choice for advanced, real-time 3D, whereas — thanks to faster CPUs and an improved Flash player — it's Flash that developers will turn to when creating 2D games. With Studio MX's Fireworks and Flash, you have all the software you need to create excellent-looking and potentially complex games.

Part 1: Designing the site in Flash

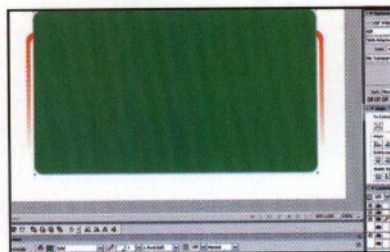
Creating the foundation for an obstacle-avoidance car game...

Shadow effect

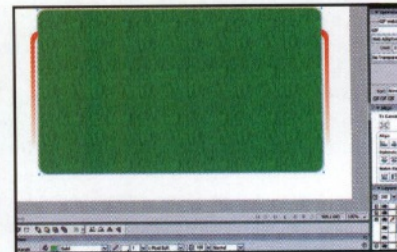
Adding a drop shadow to your objects will give them a sense of increased depth.



1 Create a new Fireworks document. Give it dimensions of around 900x600 pixels, so it's compatible with most screen resolutions. Draw a rounded rectangle, and give it a vertical gradient fill. Overlay a smaller rounded rectangle, then select Modify > Combine Paths > Punch.



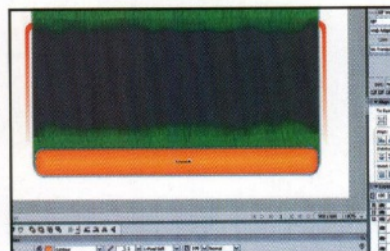
2 The Punch command is useful for creating borders like the one you should now have in front of you. Next, add another rounded rectangle, this time with a dark green fill, a white border and a 0-length drop shadow. Still in the Properties panel, add a 30 per cent sand texture.



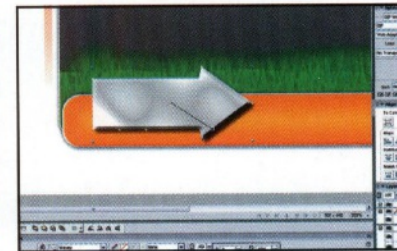
3 Copy the rectangle you've just made, and paste it on top of the old one. In the Properties panel, remove the drop shadow and outline, and replace the dark green fill with a lighter one. Add a 100 per cent Grass texture, and check the Transparent box.



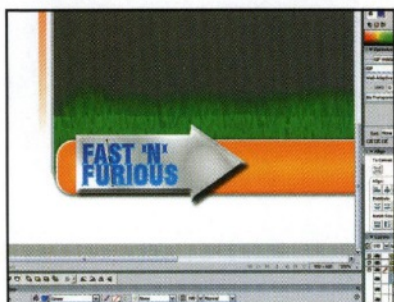
4 You should now have a grassy-looking rectangle. Now we need some road for the car to race on, so draw a grey rectangle the same width as the previous ones. This time, give it a texture resembling tarmac — something like 40 per cent Wood — then use the eraser tool to wear away the edges.



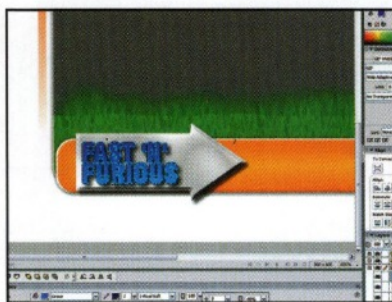
5 To create the yellow footer as above, draw another rounded rectangle, this time with a contour gradient, mixing yellow and orange. Give it a small rounded bevel and subtle drop shadow in the effects portion of the Properties panel. In addition, add a two-pixel soft outline.



6 To create the game's logo, select the arrow tool from the vector toolset. Draw a fairly large arrow, and apply Wave gradient, mixing grey and white. In the effects box, add a medium-height square bevel and a drop shadow. The resulting shape should now look fairly metallic.



7 Using the text tool, type your game's title over the arrow, then choose Text > Convert to Paths. Ungroup, and choose Modify > Combine Paths > Union. You will now be able to apply a bold gradient fill to the entire body of text, as above.



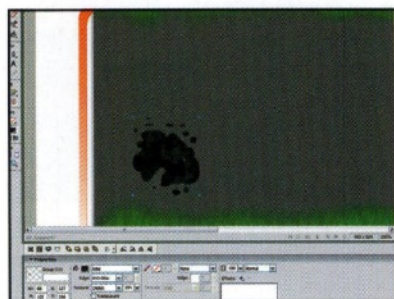
8 Give your text a one- or two-pixel outline, then add a drop shadow in the same direction as the underlying arrow. Now we need to skew both the arrow and the text to give it that 3D feel, so select both, then choose the distort tool from the toolset.



9 Distort the items so that one side seems to be further into the distance. Next, choose the bevelled rectangle tool from the vector toolset and draw a red shape over the right side of the footer bar. Give it a bold white outline, and type in 'number of collisions' followed by a number.

Do your homework

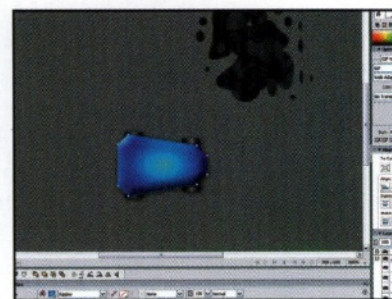
As your ActionScript projects become larger in size, make use of the Include command: this allows you to keep track of your code in a much more organized way.



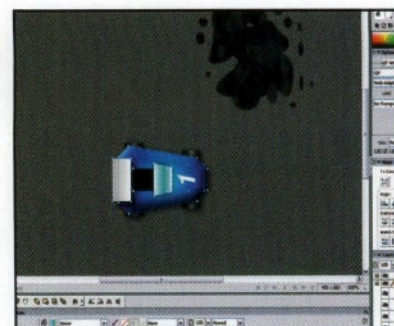
10 To create the oil slick obstacle, draw a number of overlapping oval shapes with the ellipse tool. Select them all, then choose Modify > Combine Paths > Union. Once the shape is combined, fill it with black, then apply a 25 per cent Oillick texture.



11 For the tyre obstacles, use the Doughnut tool, again from the Vector Toolset. One you've drawn a respectable-looking tyre shape, apply a three-part radial gradient fill, with black at either edge and grey in the middle.



12 To create a decent-looking car graphic, start with a bevelled rectangle, and use the Sub-selection tool to bring in the front two corners. Give it a rectangular gradient fill, and apply a subtle bevel in the Effects box. Draw the wheels by applying a three-part gradient to a rectangle.



13 Add some finishing touches using rectangles to create a spoiler and windscreen and tweaking the shapes with the Sub-selection tool. Once you've finished with your embellishments, add another subtle drop shadow. Finally, make sure to group all the obstacles' individual pieces for ease of export.



14 Save your document, then select all items apart from the obstacles, the car and the collision text. Copy and paste everything else into a new window. Choose Modify > Canvas > Trim Canvas. Now all you need to do is to save this new background graphic.

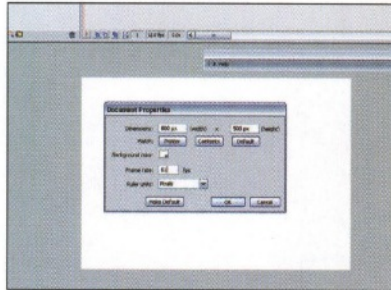


Part 2: Bringing the game to life

Animating your game with Flash...

Limitation

Flash's performance can struggle if too many of your BMP's are overly large and complicated.



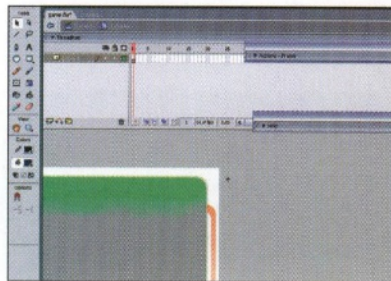
1 Open Flash MX 2004, and create a new Flash file with dimensions of 800x500 pixels. Select a suitably high frame-rate – we're using 61 frames per second in our example file. Rename the first layer 'actions' and leave it blank – we'll write our ActionScript code here later.



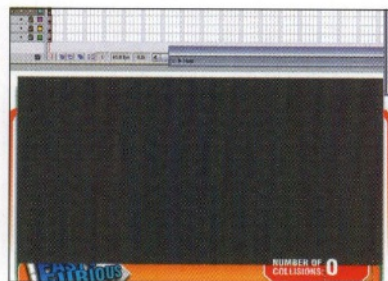
2 Create a new layer, choose Import > Import to Stage, and select the background graphic that we saved in Fireworks. Once it appears on stage, centre it. Write some text saying Number of Collisions, and place a dynamic text field next to it, with the variable name collisions.



3 Next, create a new MovieClip (Insert > New Symbol > MovieClip), and call it Car. We'll leave it blank for now, but place it on a new layer on the stage, with the instance name car. On the layer above this, create another blank MovieClip, this time with the name Obstacles.



4 Again, give the obstacles' MovieClip the instance name Obstacles, and move it to just right of the stage, at Y position 24. Double-click it to drill down to its individual timeline. Once inside, create another two MovieClips, called Oil and Tyres, and with identical instance names.



5 Return to the main timeline, and create a new layer called Mask directly above the layer that contains our obstacle's MovieClip. To provide the illusion of the obstacles scrolling past the car, we'll use a Mask layer. Draw a rectangle of the same width and height as the background image.



6 Right-click the Mask layer, and select the Mask from the menu. The layer below it should now be indented, and green masking icons should have appeared in both layers' names. We'll return to Flash later on, to drop in our game elements (the car and obstacle graphics).

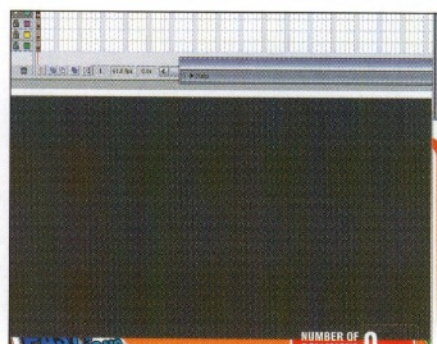
Resources: check out these sites for more on Flash and Fireworks



Macromedia Fireworks Developer Centre

Get the latest tips, tutorials and extensions from leading industry professionals at Macromedia's own Fireworks developer network site.

www.asp.macromedia.com/devnet/mx/fireworks



Prototype

A massive selection of extensions to the existing Flash objects – this site is a fantastic resource for nifty ActionScript routines.

www.proto.layer51.com



WaxPraxis

Flash programmer Branden Hall's Weblog. This is a useful place to visit, and is aimed at intermediate to advanced developers looking to enhance their knowledge.

www.waxpraxis.org

Part 3: Moving objects

Coding the control mechanism and obstacle movement...

Control of our game is going to come via the keyboard's arrow keys. Receiving and manipulating keyboard input is not exactly one of ActionScript's fortes – there are plenty of methods associated with the keyboard, but coming up with a flexible solution can often be quite hard. The trick is to use the event model, and assign a listener object to receive and handle the input.

We're also going to need to write a function that will send obstacles towards our car at regular intervals. This means using the `setInterval` command in conjunction with that age-old chestnut, `duplicateMovieClip()`.

Moving the obstacles requires some equation-based tweening functions, so for the purposes of this walkthrough, we're going to focus on the code required to actually create and deliver the obstacles. The code responsible for the movement is included on the cover disc as an `ActionScript` file.

1 Lines 1–2: Use the 'library.as' file to emphasise the car's movement.

```
stop();  
#include "library.as"
```

2 Lines 3–7: Set up a listener object and begin a function that's called when a key's pressed. Halt any existing script within the car, then set up a couple of movement variables.

```
listener = new Object();  
listener.onKeyDown = function
```

```
() {  
    delete car.onEnterFrame;  
    var yval = 0;  
    var xval = 0;
```

3 Lines 13–20: Okay, leak the load-
ingtxt MovieClip invisible and we
enable the img clip itself.

```
clip.myLoader = new
    MovieClipLoader();
clip.myListener = new
    Obj+thumbs[i].
    attributes.thumb,clip.empty);
```

4 Lines 8–19: Use a switch statement with the `Key.getCode()` method to see which key's pressed. Compare the key's character code to Flash's native arrow key values, and set the movement values. Create an `onEnterFrame` function to moves the car in the specified direction.

```

switch (Key.getCode())
{
    case Key.DOWN: yval = 8;
    car.rotate(15,30); break;
    case Key.UP: yval = -8;
    car.rotate(-15,30); break;
    case Key.LEFT: xval = -8;
break;
    case Key.RIGHT: xval = 8;
}

car.onEnterFrame = function
(){
    this._y += yval;
    this._x += xval;
}

```

5 Lines 20–24: When the key's released, rotate the car back to its original position and delete its onEnterFrame script. Assign the listener object's methods to the global Key object to listen for key presses.

```
listener.onKeyUp = function(){
    car.rotate(0,30);
    delete car.onEnterFrame;
};
Key.addListener(listener);
```

6 Lines 25–30: Use the `SetInterval` command to fire off a function called `Obstacles` every 1.5 seconds, then declare the function, which begins by creating a new instance of the tyres' obstacle, via `duplicateMovieClip()`.

```
interval = setInterval(obstacle  
es,1500);  
_root.i = 200;  
car_depth = 400000;  
function obstacles()  
{  
    duplicateMovieClip("obstac  
les.tires","tires"+_root.i,  
40000 -i);
```

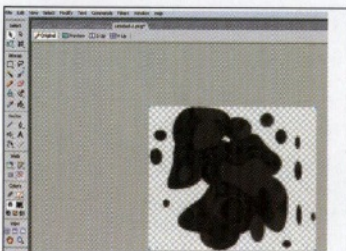
7 Lines 31–51: Position the tyres and oilslick at a random y value and rotation. Move them across the screen with Move() function. Assign a function to both to check whether they've collided with the car. If they have, remove them and incre-

ment the collisions total.

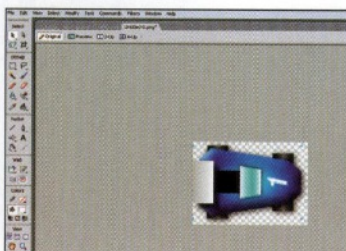
```
var tire = eval("obstacles.  
tires"+_root.i);  
        tire._y = Math.randomBetween  
(100,300);  
        tire._rotation = Math.  
randomBetween(0,360);  
        tire.Move(-1000,tire._y,  
250);  
        duplicateMovieClip("obst  
acles.oil"+_root.i,i);  
        var oilclick = eval("obst  
acles.oil"+_root.i);  
        oilclick._y = Math.random  
Between(100,300);  
        oilclick._rotation = Math.  
randomBetween(0,360);  
        oilclick.Move(-  
1000,oilclick._y, 250);  
        tire.onEnterFrame = oilclick.  
onEnterFrame = function(){  
        if(this.hitTest(car)){  
        delete this.on  
EnterFrame;  
        this.remove  
Flicker(10);  
        _root.collisions ++;  
        }  
        }  
        tire.onComplete = oilclick.  
onComplete = function(){  
        this.removeMovieClip();  
        }  
        _root.i++;  
};
```

Part 4: The game elements

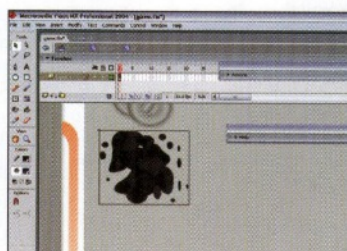
Adding the car and obstacle graphics...



1 Return to your original master Fireworks PNG file. Once open, select and copy your oilslick graphic. Create a new document with a transparent background, then paste the oilslick in. Trim the canvas to eliminate any extra surrounding space, then save as a PNG.



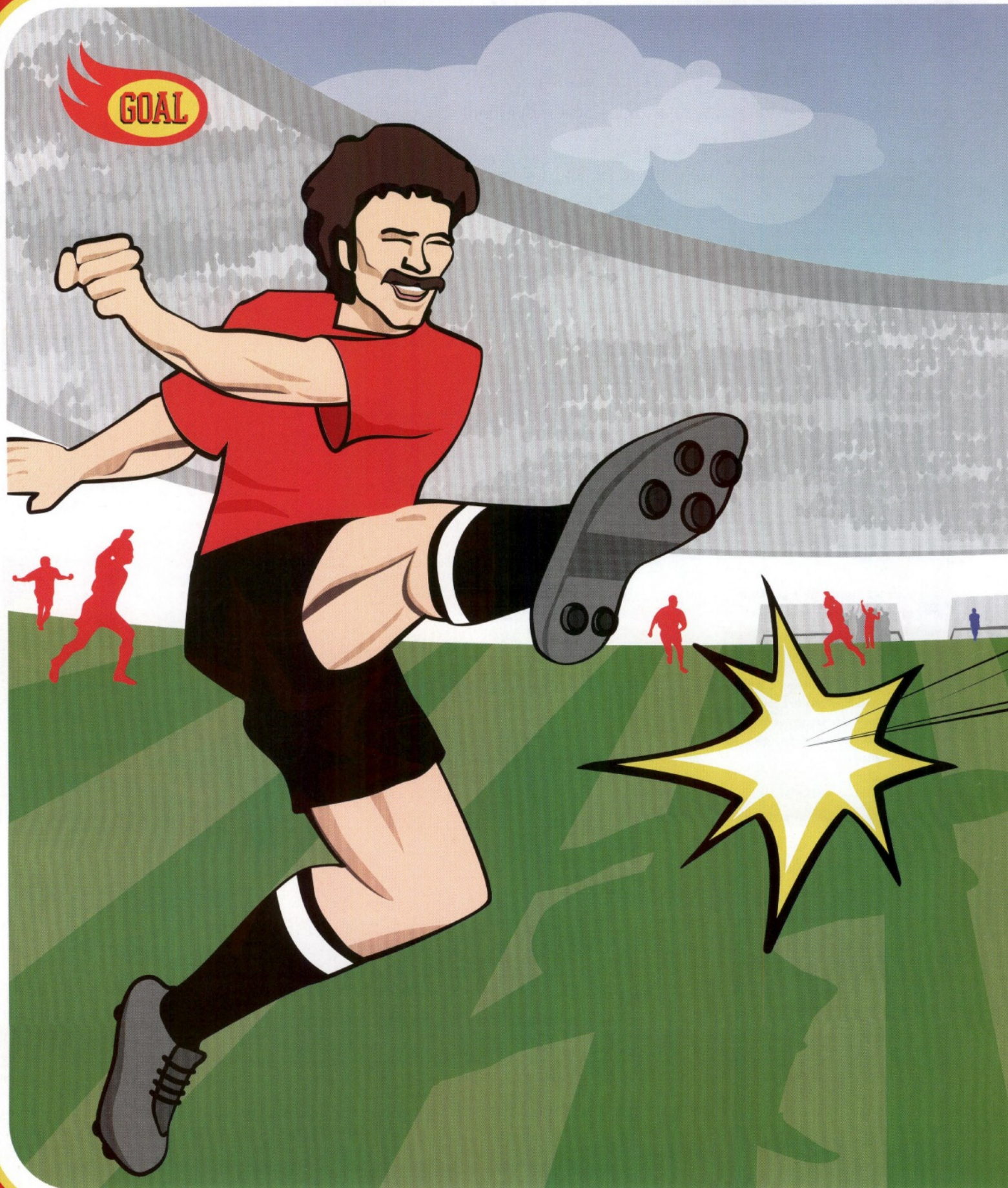
2 Now repeat the previous step for the other two objects – the car and the tyre. You should now have a separate PNG file for each of the three objects. Open Flash MX 2004 again, and bring up the game interface file we worked on before.



3 We previously created blank MovieClips that acted as placeholders for the various objects. For each one of these (the car, the tyres, the oilslick), drill down to their individual timeline, choose **Import > Import To Stage**, and select the corresponding graphic.



4 Save your file and choose File > Publish Preview > Flash to see the game in action. Obviously, there's extra functionality that you could add. For example, how about some code that prevents the car from leaving the boundaries of the layout?





Create winning games in Flash

Flash MX 2004 even sounds like a football tournament (well, the 2004 bit does), so what better software to use for creating a penalty shootout game...

Taking a penalty is all about practice, composure and confidence. Creating a Flash game also takes practice, and you need to have a plan and then the confidence to see it through. Games are multi-skilled affairs combining graphical assets and code. Over the next few pages we show you how to create the graphics, setting this all up in a Flash movie and coding the game.

For asset creation we'll be using Poser 4 and LightWave 3D. Although traditionally thought of as a vector package, Flash can handle bitmaps very well and 3D animation is quick and relatively easy to edit. Flash is a flexible package enabling the developer to choose the best way to set up the assets. This is both a benefit and a curse — with experience, you get into a style of setting up a game, but for a beginner, deciding how to structure the project can be one of the biggest headaches. ActionScript coding is great fun, easy enough for the beginner and structured enough for the seasoned programmer.

So whack in the cover disc, navigate to the source files, boot up the computer and follow along on the step-by-step program to ball-skill heaven.



Expertise provided by Flash and Director guru Nik Lever. You can contact him on nik@catalystsystems.com

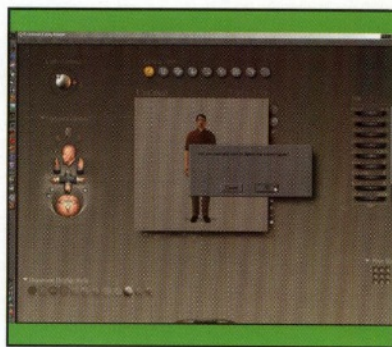


All the components needed to complete this tutorial can be found on the cover disc.

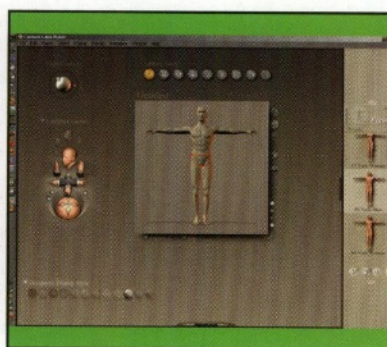


Part 1: Creating the assets

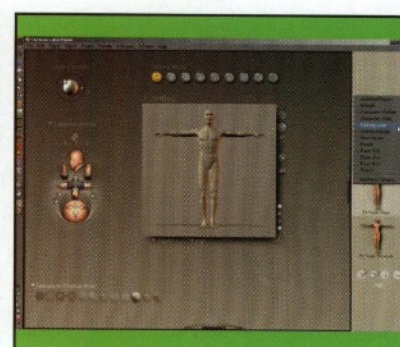
Sprite graphics can be created easily using Poser and LightWave 3D...



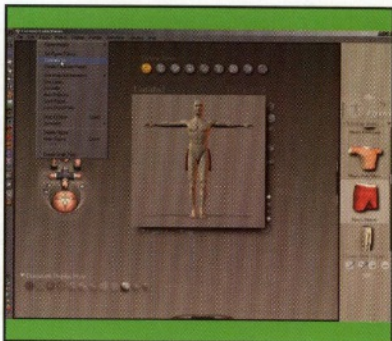
1 To create the player and goalkeeper we're going to use a combination of Poser 4 and LightWave 3D 7.5. First, run Poser 4 and select File > New. This creates a default Poser scene featuring the default male character. Select the hips of the man by clicking on them, and press Delete.



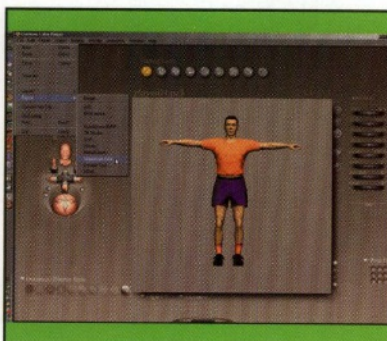
2 You should now have an empty project. Click the side bar device at the right and select P4 Nude Man from the icons. Using the buttons just below the icons, click the one with two ticks – Create New Figure. This adds the nude figure to the project.



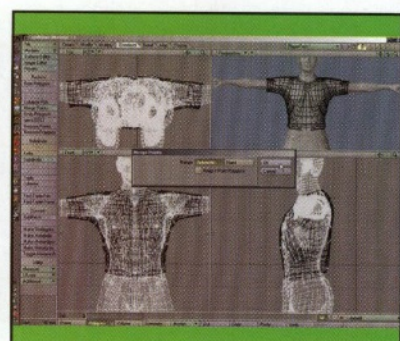
3 Again using the side bar, click on the word People and from the menu select Clothing-male. You need to add shorts and a shirt to the character, so select Men's shorts and again click Create New Figure. Repeat this process to add your choice of shirt.



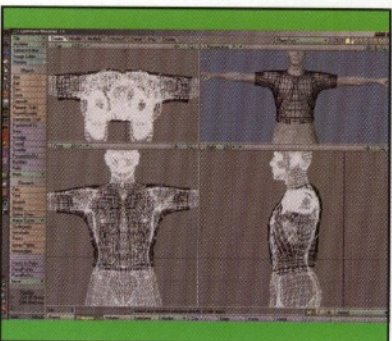
4 At this stage, the clothes will not conform to the character – you must tell Poser that this is what you require by using the menu option Figure > Conform To... This brings up a dialogue box where you select which object you want the clothing to fit. Select Figure 1 and click OK.



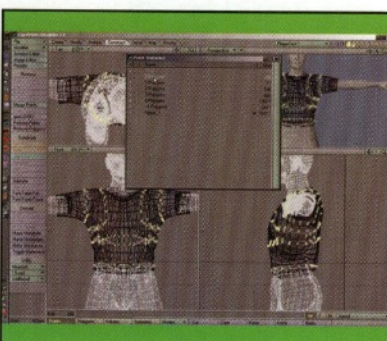
5 You might find it quicker and easier to open the file player.pz3 from the cover disc. You should now have a completed figure to export to LightWave. Choose File > Export > Wavefront OBJ. In the next dialogue, select Single frame and save using the default settings.



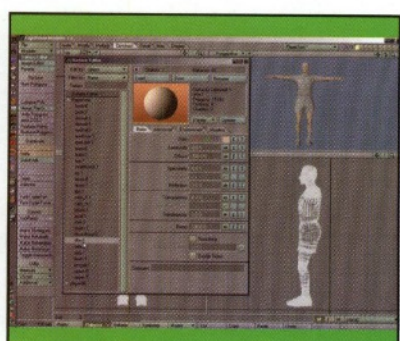
6 You'll need to do some 'tidying-up' before you can use the model for animation. Start by removing unwanted points. Select Construct > Merge Points. Select Automatic and unselect 'Keep 1-point polygons' then press OK.



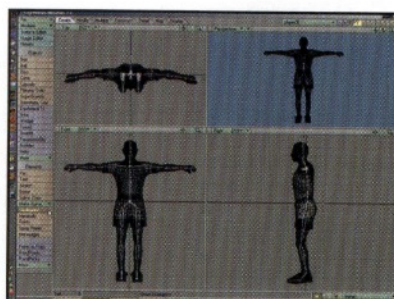
7 Click on the shirt and press X to cut it out, then move to an alternative layer and press v. Select layer 1 as the foreground and layer 2 as the background, highlight the torso polygons beneath the shirt and press delete to remove them.



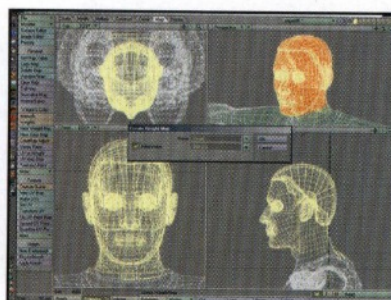
8 Having deleted the torso polygons, you might find you have stray points. Select Point selection mode at the bottom of the screen and press w to bring up the Point Statistics window. Click + to the left of 0 Polygons, and press Delete.



9 Repeat Steps 7 and 8 for the polygons under the shorts, boots and socks. Then use the Surface Editor to set the colours and texture detail. The Export/Import process retains surface names but loses the surface detail.



10 To speed things up, load in the object 'player.lwo' from the folder on the cover disc. Here is the completed player. In layer 2, create the bones to animate the character using Create > Skelegons. Name each bone sensibly using Detail > Rename Skelegon.



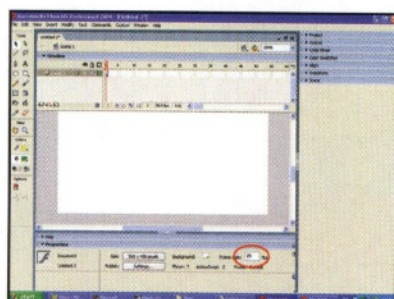
11 The points a bone affects can be assigned using weight maps. Select a number of vertices and then use Map > New Weight Map. Name the map. Choose Weight Map Shade from the Perspective window dropdown menu to view the weight maps. Save the object then move to the animation and rendering application: Layout.



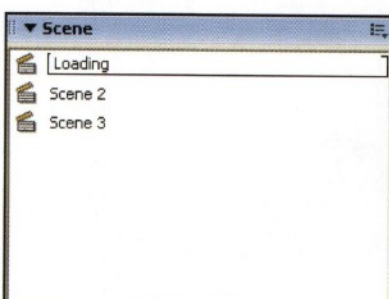
12 Select Items > Add > Bones > Convert Skelegons to Bones. Select the bone button at the bottom to move parts of the player's body. Use the keyframe tools to animate; press L to create a limited render region. Drag the dotted yellow box till it forms the minimum area for the player's movement. Render the scene saving as 32-bit pngs.

Part 2: Organising the assets

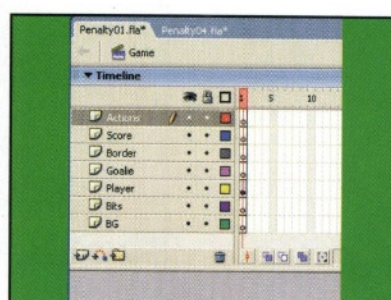
If you're new to Flash, deciding how to structure your game will be one of the most difficult decisions...



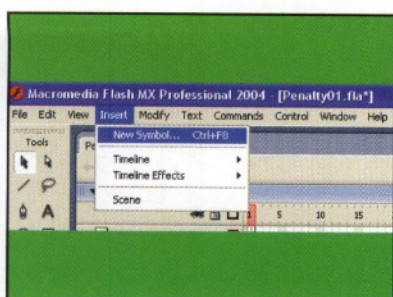
1 Create a new document by choosing File > New. Then select Flash Document from the General tab of the next dialogue box. Open the Movie Properties window by selecting Window > Properties and then clicking in an empty area of your Movie. Set the framerate to 20 fps.



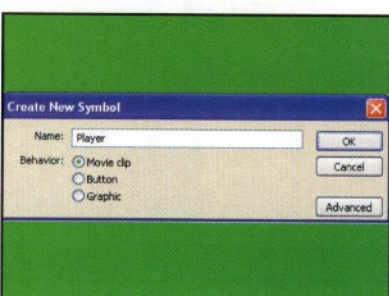
2 A default document has a single scene but we need three. So choose Insert > Scene twice to add two extra scenes. Then open the Scene Window by selecting Window > Design Panels > Scene. Using the Scene Window you can rename the Scenes to Loading, Game and Game Over.



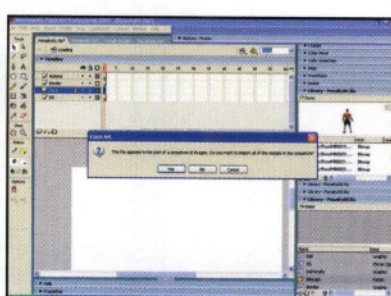
3 Loading and Game Over scenes will have the layers BG, Text, Border and Actions. The Game scene uses the layers BG, Bits, Player, Goalie, Border, Score and Actions. Add layers by clicking the bottom-left icon in the Timeline Window. Rename the layer by double-clicking the name.



4 Player, Goalie, PowerBar, Ball and BG are all MovieClips. A Flash MovieClip is a separate movie that contains its own timeline; multiple timelines give Flash its power as a development tool. To create a MovieClip, select Insert > New Symbol... and choose Movie Clip.



5 Create a new MovieClip and name it Player. Newly created symbols will appear in the Library. Choose Window > Library to open the Library window. Then drag the Player clip onto the stage. Double-click the dot that indicates an empty MovieClip.



6 We want to add the images we created at Step 12 (above) to this clip. Select Import > To Stage... and locate the first image in the animation. Flash will spot this as a sequence. In the following dialogue, choose Yes to 'Do you want to import all the images in the sequence?'

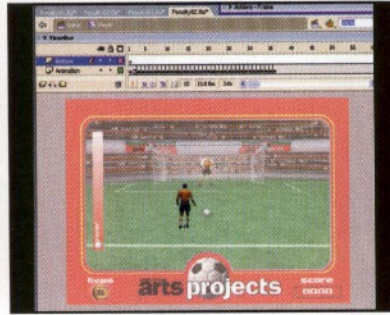


Part 3: Adding the code

It's time to start turning the graphics into a game using ActionScript.



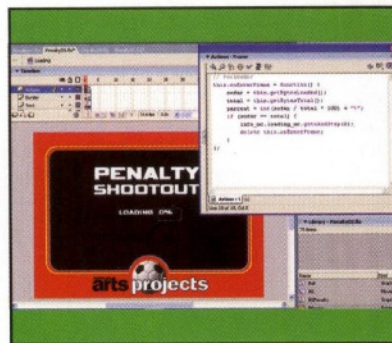
1 The background, ball and goalkeeper are also LightWave models. Import these in the same way, making each one a separate and named MovieClip before you begin to work on the code. The final step in creating the assets is to draw the borders and score graphics.



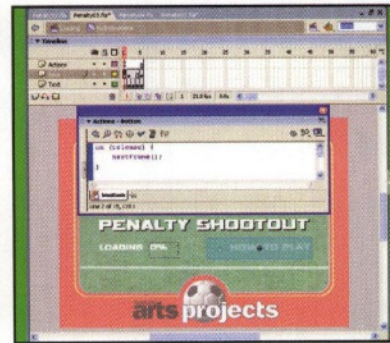
2 Penalty02 fla on the cover disc is a Flash document containing all the assets but no code. The first step is to create animation loops inside the various MovieClips. The Player clip has two actions: Shuffle, for moving from side to side; and Shoot, to kick the ball. Add an Actions layer.



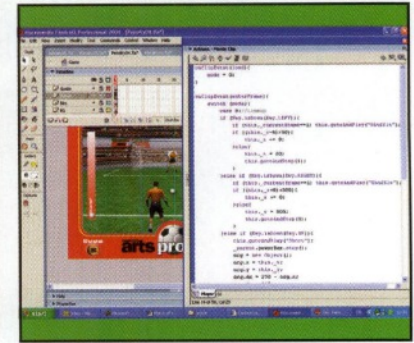
3 Add a Shuffle frame label to frame 2, and a Shoot label to frame 11. Click on frame 2, and in the Properties window type Shuffle in the Frame Edit box. Shuffle is a looping animation, so go to frame 10 and press F7 to insert a blank keyframe. Press F9 to open the Actions window and type: gotoAndPlay ("Shuffle");



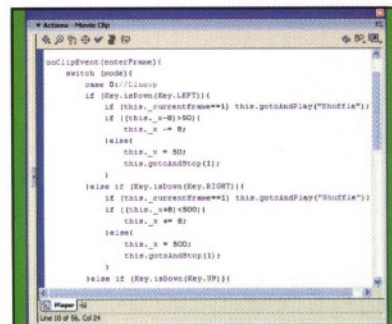
4 The indicator to show how much has been downloaded in this game is handled by using an onEnterFrame script defined on the Actions layer frame 1. This derives a percentage loaded by dividing the bytes downloaded by the total number of bytes in the swf.



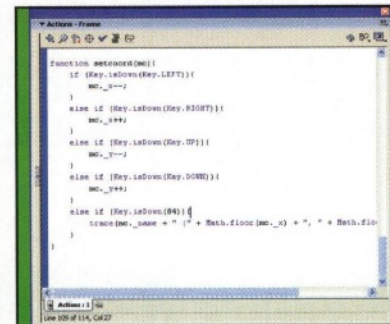
5 The Loading scene also contains the instruction. The script for the 'How to Play' button can be seen by opening Penalty03 fla from the cover disc, going to the Loading scene, double-clicking the turquoise area on frame 2 and looking at the Actions window.



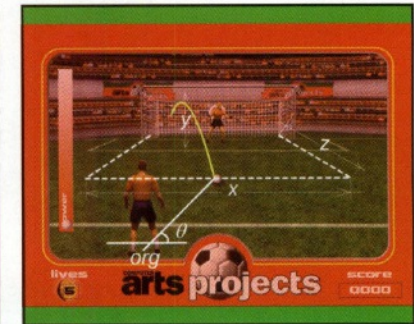
6 Player motion is handled using an onClipEvent (enterFrame) handler. This event uses the Key object to detect key presses. To view the code, open Penalty04 fla from the cover disc, select the 'Game' scene and click on the player. The script will be seen in the Actions window.



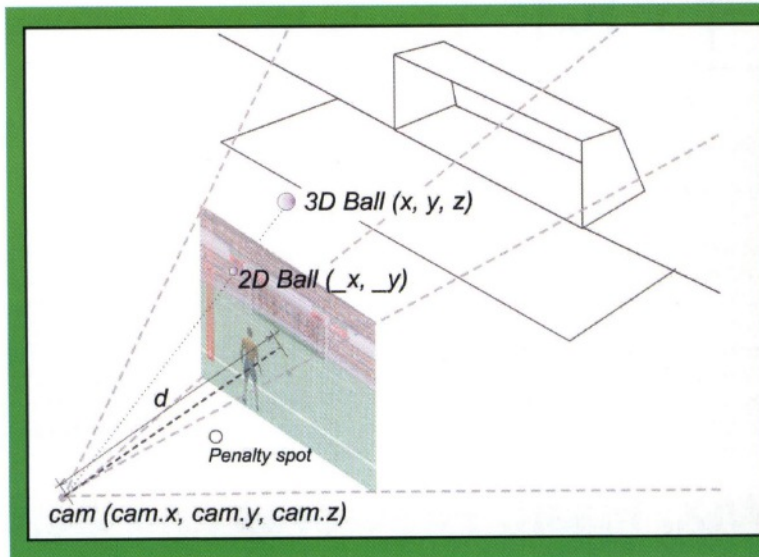
7 There are three possible mode settings for the player: 0 for line up, 1 for the run up, and 2 after the kick. The script checks the Left, Right and Up arrows if the current mode is 0. Left and Right are used to move the player if the player will stay within bounds.



8 The setCoord function, defined in frame 1 of the Actions layer for the Game scene, is a function for moving a MovieClip. The function takes a MovieClip, mc, as a parameter and then uses the arrow keys to move it and the T key to send its current position to the Output window – useful for determining the bounds of a moving clip.



9 Pressing the Up arrow triggers the kick. Because of the viewing angle the kick is handled as a 3D problem. The kick angle is θ , and the strength of the kick is defined using the 'currentframe' of the power bar MovieClip. The problem reduces to just these two parameters, which are passed to a function 'kick' defined in 'ball'.



Adding a third dimension

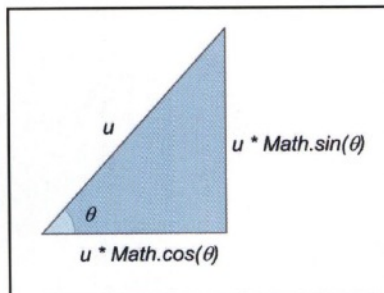
To use 3D we add a third coordinate, z , to the x and y positions. We can define the location of the goal using values of x to define left and right and values of y to define up and down. In addition, the position back from the ball kick location is defined using z . But we also need to be able to convert these 3D coordinates back into screen coordinates so we can position the ball. This is surprisingly easy. First, we need to know the camera's location ($cam.x$, $cam.y$, $cam.z$) and the distance of the camera to the screen, d . Then the location is defined as:

$$_x = ((x - cam.x) * d) / (z - cam.z) + cam.x;$$

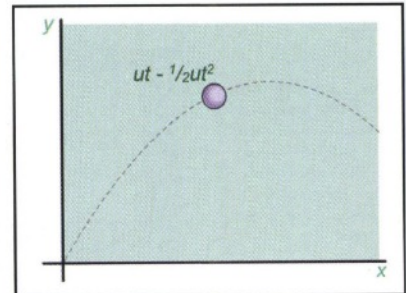
$$_y = ((y - cam.y) * d) / (z - cam.z) + cam.y;$$



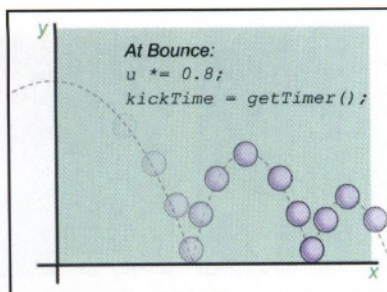
10 When using 3D-to-2D conversions in this way, the position of the camera is crucial to the success. The function `drawRect` defined in the first frame of the Actions layer is designed to draw a rectangle using the 3D conversion. This needs adjusting until it matches the ground perspective.



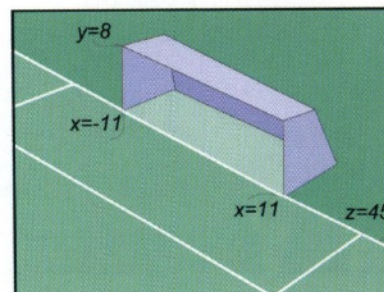
11 The ball is moved in the x and z in 3D space for each frame, using the variables dx and dz calculated in the kick function. They are calculated from the angle of the kick based on simple trigonometry. The side near the angle uses `Math.cos` and the side opposite `Math.sin`.



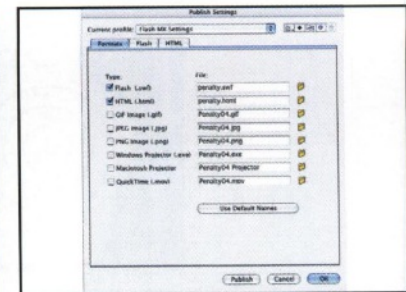
12 Projectile motion takes a launch power ' u '. The height of the ball at time ' t ' is defined as $ut - 1/2gt^2$ where g is gravity. To convert this into useful values for your game takes practice. Adjust the time taken to start falling as indicated in code and adjust the speed of the fall using the value for gravity.



13 The advantage of using 3D for the ball location is that the problem is made easier. A value of y of zero means we need to bounce the ball. Which involves reinitialising '`kickTime`' and then simply reducing u , dx , and dy by 20 per cent, (by multiplying them by 0.8).



14 The ball reaches the goal when the z value is 45. A goal is successful if $-11 < x < 11$ and $0 < y < 8$. This is tested in the `onClipEvent(enterFrame)` handler for the ball. Then a new penalty is initiated using the `newPenalty` function defined in frame 1 of the Actions layer.



15 Save your document, publish the SWF or both SWF and HTML by going to `File > Publish Settings` and there you have it – a 3D game that uses a bit of mathematic theory in 36 steps! ■



Studio MX: Shoot-'Em-Ups



All the components needed to complete this tutorial can be found on the cover disc.

Make a shoot-'em-up game with **Tom Knowles**.
Do ya feel lucky? Well, do ya punk?

Making games is something most Flash designers will have had a shot at. And the chances are most of them will have made the same sort of mistakes at some point. Game development can be tricky.

That's not to say that making games is hugely difficult – it's not. What takes a bit of getting used to, however, is the type of

thinking you need to do beforehand. The main problem is that games aren't linear. Even the most avant-garde web sites have a degree of linearity: you're going to be browsing information and to do that you'll be going from one section to another. Games are different; you're giving almost total control to the user, and you're leaving their progress totally up to them. What this boils down to is

that the rules you create for your environment have to be extremely stringent; in many cases you'll have several types of input and evaluation to deal with at any one time, which makes fertile soil for bugs.

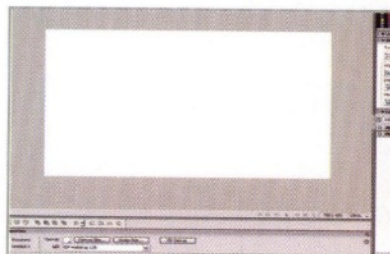
The key then is to plan carefully – think about the game rules as they apply to the game engine itself, as well as how they appear to the user.

Part 1: Use Studio MX, Flash, Fireworks and ActionScript to create your game

That's a lot of programs but we're not going to over-use any of them...

Don't overdo it

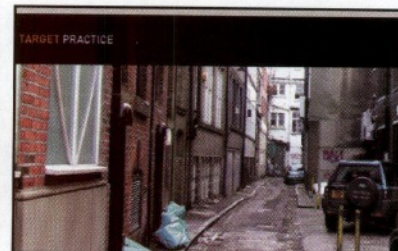
Try to keep your Flash game as simple and effective as possible – overdoing it on graphics and sound can push older computers to their limits. If you're serious about heavyweight gaming, Macromedia's Director is a more robust option.



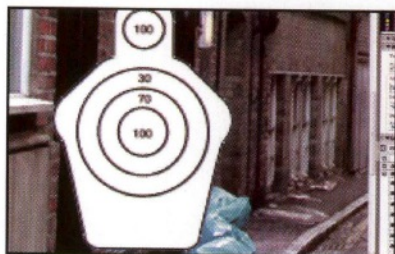
1 The first thing we need to do is open Fireworks MX 2004 and create a new document with dimensions of 750x450 pixels. The background colour isn't really important since we'll be laying graphics over it anyway. Now, since we're making a shooting game, we're going to need a suitable background image.



2 We've chosen a photograph of a Manchester alleyway, as it has a suitably grimy atmosphere. However, most digital photographs are a little washed out when they come straight from the camera. To give any image a boost, lightly increase the brightness and contrast in the Filters > Adjust Color menu.



3 Once you've adjusted your background image, paste it into your main document. Using the rectangle tool, create a black border as above, using the Align panel (Window > Align) to even things up where necessary. For layout purposes, use the type tool to draw out some text fields.



4 For the purposes of this walkthrough, our enemies are going to be reasonably simple firing range-style targets. Use the rounded rectangle tools in conjunction with the super-useful Modify > Combine Paths menu in order to get the basic shape, adding outlines and extra detail where appropriate.



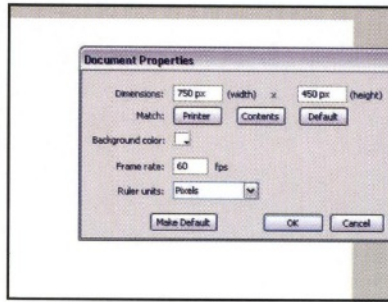
5 Drawing the crosshair/target is also a very simple affair; we've just used simple circles and rounded rectangles along with the Numeric Transform dialogue box (which allows you to make precise rotations). Be wary of making it too elaborate – Flash might slow down if it has to drag an enormously complex shape around.



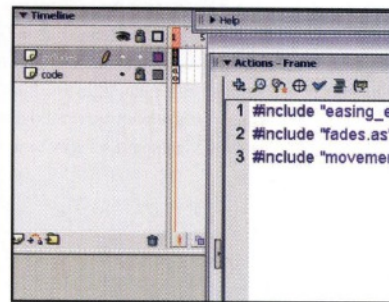
6 We can make more good use of Fireworks's vector tools when we come to draw our rather gruesome wound graphics. Build up a splodge of varying red circles. When you're done, combine them, and then punch through with another circle. For a final touch add a red glow in the properties panel.



7 With our interface drawn, we now need to prepare our graphics for use in Flash. Copy and paste the wound graphic into a new, transparent Fireworks document and save it in the usual PNG format. Next, create a new version of the layout with just the background image and black borders.



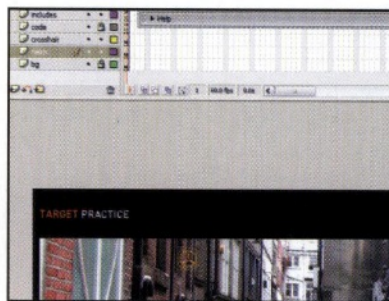
8 Save the new document as another PNG. Once that's done, keep Fireworks open and start Flash MX 2004. Create a new Flash document with a high framerate and the same dimensions as your Fireworks one, and save it, taking care to specify ActionScript 1.0 and Flash Player 7 in the publish settings.



9 Rename the first layer of your file to includes, and in the Actions Panel, type the above ActionScript code include commands, adjusting the paths if you choose to keep your ActionScript libraries in a certain place. Add another new layer below that, this time with the name code.



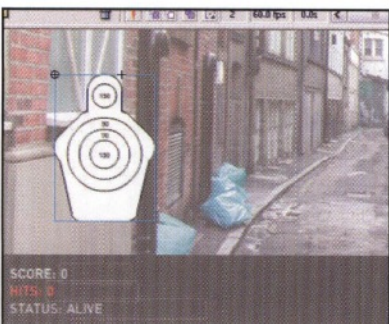
10 We'll place our ActionScript code on this layer later on, so you can leave it blank for now. Create another four layers, as above, and call them crosshair, enemies, fields and bg. On the bg layer, choose File > Import > Import To Stage, and pick the background graphic we saved earlier.



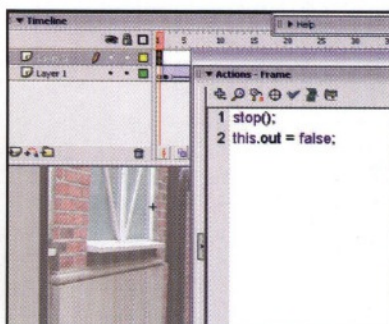
11 On the crosshair layer, simply copy and paste the crosshair we drew in Fireworks. Flash will automatically convert it to a MovieClip symbol. All we need to do is make sure it has the instance name crosshair. Now move to the fields layer, and type out the same text as we had on our layout.



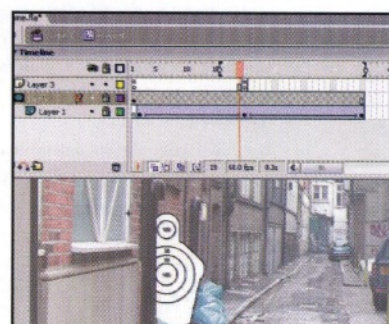
12 As above, convert the relevant fields to Dynamic Textfields, with the instance names scorefield, hitsfield and statusfield. Import your wound graphic to the library and, in its linkage settings, export for ActionScript in the first frame and give it the id wound.



13 Return to Fireworks and then copy and paste the enemy graphic onto Flash's enemies layer. Give it the instance name enemy. Next, convert it to another MovieClip, this time with the instance name enemy1. Inside this symbol, convert the enemy instance to a motion tween, and move it to the second frame.



14 Create a layer above the enemy instance, and add the above ActionScript on the first frame. On the twentieth frame, add a simple stop(); action. As above, create a mask layer above the enemy instance, mimicking a hiding point on the general game background image.



15 The idea here is that you can add as many of these individual animated enemy MovieClips as you wish, as long as you take care to give them unique instance names, and pump them into the enemyArray in our code so that they can be accessed by the game's engine.

Extend Functionality

One of the great things about using Prototype methods externally is that (as long as you don't change their API) you can return to them and extend their functionality at a later date, without affecting code that uses them already.



Part 2: Actionscript code part 1

Any good shoot-'em-up requires lots of enemies...

As we will see further on in this tutorial, Flash's event model provides some very simple and elegant functions for handling mouse movements and clicks, as well as dispensing with the arrow cursor altogether. In true old-skool fashion, our game's enemies are going to appear from behind various obstacles in the background. Optically, we've achieved this with some straightforward masking and tweened animation. The only problem we have now is how to fire off these animations. In order to provide a random pattern, we're going to need to play any one of our four enemy animations at a set time interval. This is where the `setInterval` command will come in very useful. It will allow us to specify a function for Flash to execute repeatedly in the background at a given time, giving us the opportunity to rely on real time rather than animation frames.

1 Lines 1–6: As we mentioned earlier on, we want to be able to add as many different enemy animations as possible, without having to drastically recode when we do so. So, in this snippet of code, we create a global array (available from anywhere in the movie) in which we can simply place references to all the enemy `MovieClip` instances we want to include.

```
stop();
_global.enemyArray =
new Array();
enemyArray.push(enemy1);
enemyArray.push(enemy2);
enemyArray.push(enemy3);
enemyArray.push(enemy4);
```

2 Lines 7–15: A good habit to get into is to use configuration variables. Rather than hard coding numeric values into the body of your code, it'll be much cleaner and far easier to update if you declare them at the start. Some notable examples here are the times which control how long the enemies stay 'unhidden', as well as the initial time between enemy appearances.

```
_global.minTime = 500;
_global.maxTime = 3000;
_global.mainTime = 2000;
_global.woundDepth = 6000;
_global.noHits = 0;
_global.hitLimit = 5;
_global.score = 0;
_global.status = "ALIVE";
_global.interval = set
Interval(generateEnemy,mainTime);
```

3 Lines 16–44: Let's deal with our custom mouse cursor (the crosshair `MovieClip`). When the user presses their mouse button down, we fire off a little cosmetic animation using the `elasticResize` prototype (which you will be able to find on the cover disc that came with this book). We also loop through all the available enemy `MovieClips` and see if we've clicked on one. If we have, we increase the user's score and reset the main `setInterval` command with a slightly shorter time.

```
Mouse.hide();
crosshair._x = _root._xmouse;
crosshair._y = _root._ymouse;
crosshair.onMouseDown
= function()
{
this.elasticResize(150,150,
20,10,10);
for (var i=0; i<enemyArray.
```

```
length; i++)
{
var temp = enemyArray[i];
if (this.hitTest(temp.
enemy))
{
trace("On target!");
clearInterval(temp.
intervalID);
temp.
fadePlicker(0,10,10);
temp.onComplete =
function()
{
this.gotoAndStop(1);
this._alpha = 100;
delete this.
onComplete;
}
clearInterval
(interval);
mainTime -= 50;
_global.interval =
setInterval(generate
Enemy,mainTime);
score++;
scorefield.text
= score;
break;
}
}
```

4 Lines 45–54: When the user releases their mouse button, we perform another small cosmetic animation. Notice that rather than using the traditional `Drag` command to move the crosshair around with the mouse, we use the `onMouseMove` event. This provides us with instant feedback and, coupled with the `updateAfterEvent` statement, it also provides us with a smooth replacement for the normal cursor.

```
crosshair.onMouseUp
= function()
{
this.elasticResize(100,100,20
10,10);
}
crosshair.onMouseMove
= function()
{
this._x = _root._xmouse;
this._y = _root._ymouse;
updateAfterEvent;
}
```

5 Lines 55–60: This function is called when the enemy `MovieClip` is about to hide itself. Before it does, it calls the `hit` function, which fires at the user and registers a hit. Notice how the function takes a parameter which represents the specific enemy clip calling it.

```
function hideEnemy(clip)
{
hit();
clip.gotoAndPlay(21);
clearInterval
(clip.intervalID);
}
```

6 Lines 61–70: This is the main function which is called at regular, decreasing intervals. Essentially, it generates a random reference to one of the enemy clips. It then fires that enemy's animation but, before that, it checks to see if the enemy clip is currently active. If it is, then it loops around until it finds an inactive `MovieClip`.

```
function generateEnemy()
{
for (var i=0; i<enemyArray.
length; i++)
{
var enemyNumber = Math.
randomBetween(0,enemy
Array.length - 1);
var timeDelay = Math.
randomBetween(minTime,
maxTime);
var temp =
enemyArray[enemyNumber];
if (temp.out != true)
{
trace("New enemy");
temp.gotoAndPlay(2);
temp.intervalID =
setInterval(hideEnemy,
timeDelay,temp);
temp.out = true;
break;
}
}
```


Part 3: Actionscript code part 2

Sometimes, enemies shoot back...

Handling the mouse and firing off the enemies is the first step, but we also need to provide feedback for when the user has actually been hit by an enemy, as well as informing them when the game is over. To indicate a successful hit by an enemy, we'll use the `attachMovie` command to place an instance of the gunshot graphic we created earlier at a random point on the screen. At a later stage in the game's development, it might also be a good idea to add some suitable sounds.

For the sake of this demonstration, we'll keep the game going for as long as the player doesn't go over the hit limit, and when they do, we can cancel the user's ability to interact with the enemies and crosshair. Again, if we were to develop the game further, we could add some functionality that allows the user to restart.

1 Lines 1-15: The hit function is called whenever the user is hit by an enemy, and it uses `attachMovie` to display the wound graphic we have stored in our library. See also how we use a custom Math function to position the MovieClip randomly between two points. The last line of the function calls `updateHits`, which is detailed next.

```
function hit()
{
    trace("Player hit");
    woundDepth++;
    _root.createEmptyMovieClip(
        "hitclip"+woundDepth,
        woundDepth);
    var randX = Math.
        randomBetween(10,590);
    var randY = Math.
        randomBetween(100,250);
    var temp = eval("hitclip"+
        woundDepth);
    temp.alpha = 0;
    temp.attachMovie("wound", "
    woundclip", temp.
    getNextHighestDepth());
```

```
temp._x = randX;
temp._y = randY;
temp.fadeFlicker(80,10,20);
updateHits();
}
```

2 Lines 16-26: `updateHits` increments the global `noHits` variable, as well as displaying its value in the `hitsfield` textfield. It then checks to see whether the user has reached their maximum number of hits. If they have, it calls the `gameOver` function before declaring the user dead.

```
function updateHits()
{
    noHits++;
    hitsfield.text = noHits;
    if(noHits >= hitLimit)
    {
        gameOver();
        status = "DEAD";
        statusfield.text
        = status;
    }
}
```

3 Lines 27-37: Finally, we come to the `gameOver` function. This, essentially, halts any of the gaming activity by clearing the main interval function and then deleting the crosshair's mouse events. The custom Math extension in this snippet is a particularly useful piece of code that mimics the functionality of the old Flash 4 `random()` function, which was much more useful than the current version.

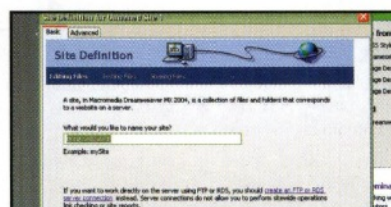
```
function gameOver()
{
    trace("Game over");
    clearInterval(interval);
    delete crosshair
    .onMouseDown;
    delete crosshair.onMouseDown;
    crosshair._visible = 0;
}
Math.randomBetween =
function(lower,upper) {
    return (lower + Math.
    floor(Math.random()*
    (upper-lower+1)));
};
```

Part 4: Publish your game online

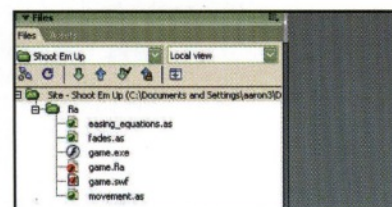
Time to let someone else have a go...



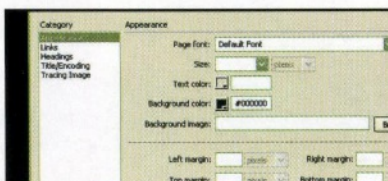
1 With our game complete, it's simply a matter of publishing your movie. Flash itself provides plenty of publishing options, but Dreamweaver cannot be matched for its organisational abilities. Using Dreamweaver to manage your site is well worth considering, especially if your project contains a large number of files.



2 Start up Dreamweaver MX 2004, and in the splash panel, choose Create New > Dreamweaver Site. You'll be taken through a wizard, asking you a number of questions. You can ignore the questions pertaining to server-side technologies and server FTP details – they're unnecessary for this particular project.



3 Make sure to pick the folder where all your development files are stored. Dreamweaver then treats this folder as an addition to its list of sites, which is displayed in the Files panel. This differs from the basic folder approach in that it allows you to use some of the tool's rich management facilities.



4 Once your site is set up, choose Create New > HTML page, and save it as `index.html`. Give it a suitable title (either in the HTML code itself or in the Title field). In Design view, move to the properties panel, and click Page Properties, changing the page background colour to black.



5 In the Files panel, navigate to your game's .swf file, and simply drag it onto your document. Ensure its path doesn't have a preceding backslash (ie it's relative to the document rather than the document root), and then save your file. Choose File > Preview In Browser to see the end result.

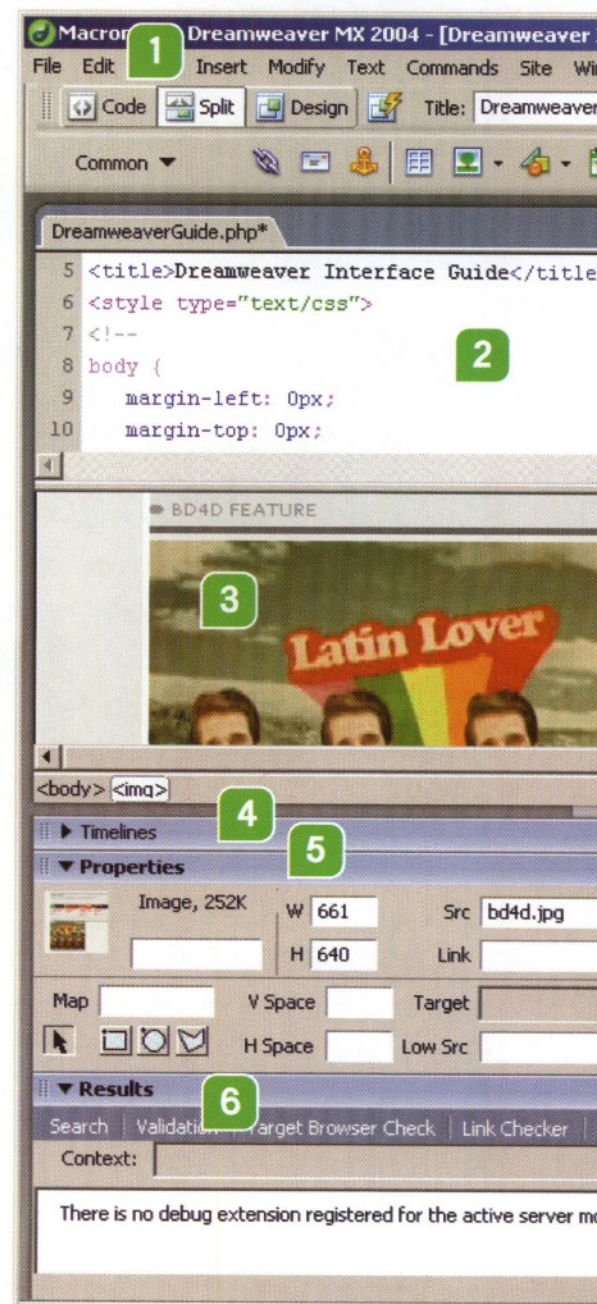


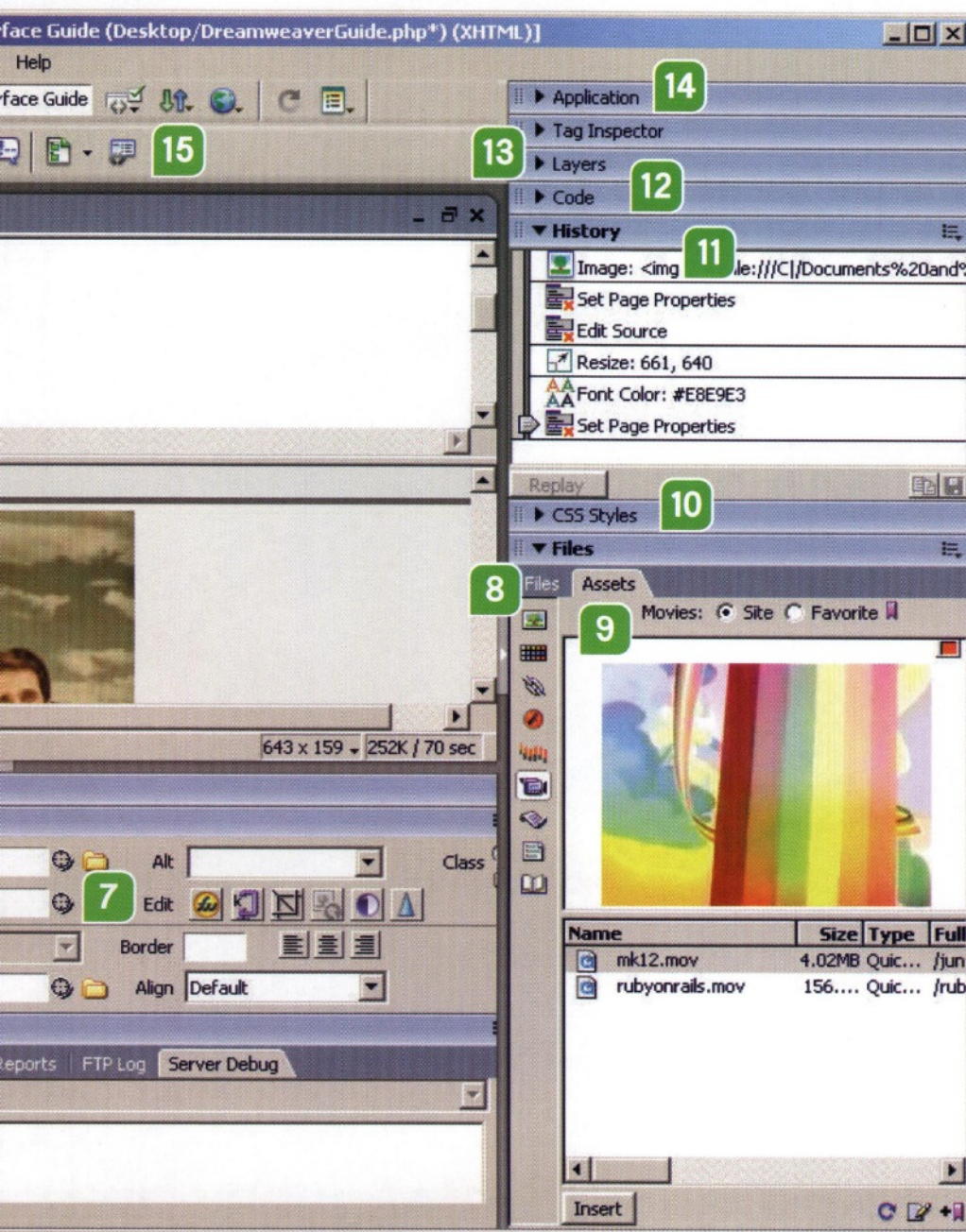
6 That's all there is to it, although if your game's file size is on the heavy side, it's definitely worth revisiting your .fla file to add a preloader. You can either build your own from scratch, or simply bolt on the default `ProgressBar` component. Doing this will enable you to place your game online with confidence.

Understanding the Dreamweaver interface

Dreamweaver MX 2004 is packed with powerful features. It gives you the flexibility of creating web sites or online applications the way that suits you best – by using its WYSIWYG interface, hand coding or a combination of the two. With built-in cross-browser validation, inline image editing, code hinting and more, it's full of features which will make your life easier...

- 1 Standard Menu**
This menu gives you all the basic functionality you would expect, such as copy, paste, save and more.
- 2 Code View**
This area enables you to hand-code XHTML, ASP, ColdFusion, PHP, and so on. It's useful for creating code for developers quickly and it now has handy features such as automatic code completion.
- 3 Design View**
This area gives you a general idea of how your page will look in a browser. You can edit the page through this WYSIWYG facility.
- 4 Timelines Panel**
This panel helps you use JavaScript to create animations on your HTML page. When you use this panel, Dreamweaver creates JavaScript for you.
- 5 Property Inspector**
Used for controlling properties of elements on the page. If you select an image, for example, you can set the height, width and other properties.
- 6 Results Panel**
This is a powerful panel that enables you to check the validity of your XHTML. It also enables you to check your file for browser compatibility.
- 7 Inline Image Editing**
Dreamweaver MX 2004 enables you to edit images right inside Dreamweaver. This includes cropping, changing the hue, and so on.
- 8 The File Browser**
This panel enables you to open files quickly from a site that you have defined on your local/remote computer. It makes opening files a lot quicker. You can also FTP a file directly from this panel.





9 The Assets Tab

This is a powerful way to access resources or assets for your web site quickly. You can actually view assets such as QuickTime, Flash, JPEG and more through this tab.

10 CSS Styles Panel

This panel helps you to manage all of your CSS styles for the current document. You can edit the styles from this panel easily.

11 History Panel

This is similar to the History feature of Photoshop or other visual applications. It enables you to go 'back in time' to a previous action.

12 Code Panel

This panel has a huge number of pre-made code snippets for you to use, including JavaScript, Meta tags, footers, headers and more. You can also add your own snippets for future use. The Reference tab includes a huge amount of helpful information from Wrox, O'Reilly and Macromedia.

13 Layers Panel

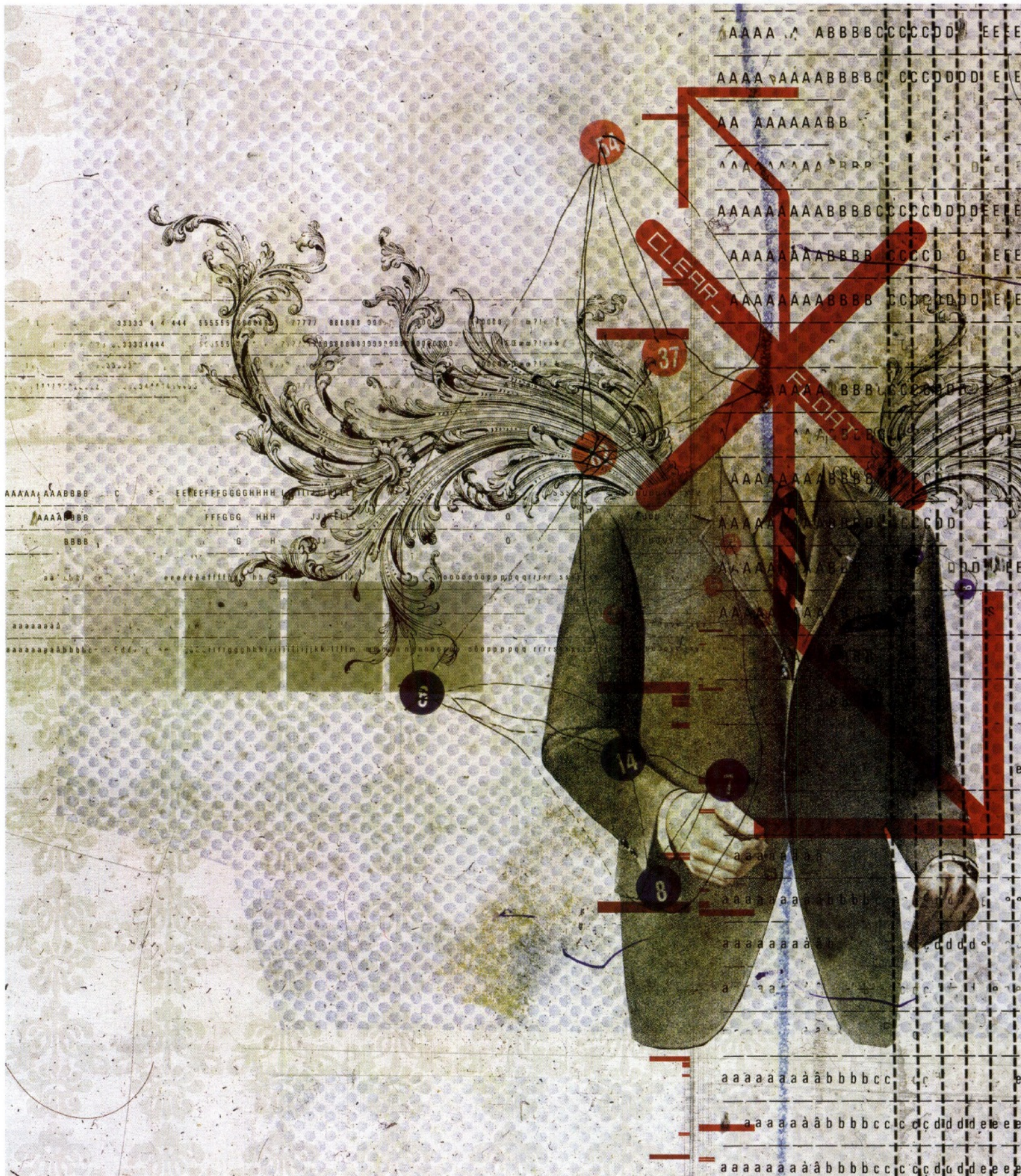
The Layers panel is a way to manage the layers in your document. Use this panel to prevent overlaps, to change the visibility of layers, to nest or stack layers, and to select one or more layers.

14 Application Panel

This panel includes the Databases, Bindings, Server Behaviours and Components tabs. These are handy ways of setting up a test server for local testing of PHP, ASP, ColdFusion and other development projects.

15 Insert Bar

The Insert bar contains buttons for creating and inserting objects such as tables, layers, and images. When you roll the pointer over a button, a tool tip appears with the name of the button. ■



CSS-based layouts

Craig Grannell unravels the mysteries behind two vital CSS properties, Float and Clear, so that you can create future-proof, modern and compliant layouts in Dreamweaver MX 2004

The Web Standards Project (www.webstandards.org) has battled with browser and web design software developers in an attempt to encourage them to support W3C (World Wide Web Consortium) standards. As a result, it's now possible to author to web standards straight out the box using software such as Dreamweaver and create sites that are still compatible with current web browsers. The next hurdle is to convince web designers to use the standards. But many continue to use tables for layout, considering CSS too "techie" and unintuitive.

However, once the basics of CSS are understood, the advantages become perfectly clear. The modular nature of CSS layouts enables concepts to be reused; presentational aspects (padding, margins, backgrounds) are defined in an external file. You can amend an entire site's design by uploading one file; and CSS encourages a semantic, logical structure, making layouts much more accessible than their tables-based equivalents.

The CSS Float and Clear properties are important for CSS-based layouts. The Float property enables you to set the float direction of any web element, so that content can wrap around it. Instead of nesting tables to create a pull quote, you can float the relevant element, controlling margins and other presentational aspects via CSS.

This will be the subject of our four-step mini-tutorial.

Float and stack

Float consecutive elements in the same direction and they "stack", enabling you to create columns in CSS by floating a number of DIV elements. Subsequent content still wraps around these elements, but by using Clear it's possible to shift content downwards until it lies below the bottom edge of the floated content. Our six-step mini-tutorial explains this process in detail. Alongside the four-step mini-tutorial, this will help you create a layout that can be adapted for countless web sites.

Layouts are largely based on styled DIV

elements. DIVs are added via the Insert bar's Insert DIV Tag button (found in the Layout section), in the subsequent dialogue box where a unique ID is assigned, and the insertion point specified (before, after or within an existing tag, or at the cursor point). Styles are created via Style > Manage Styles in Properties, Edit Style Sheet > New and then the New CSS Style dialogue. Property values are defined in the CSS Style Definition dialogue.

This tutorial assumes prior knowledge of Dreamweaver. Before you start, create a new site with a default XHTML document and attach a blank CSS file to it using the Import option (see CSS declaration). ■



Expertise provided by Craig Grannell. You can contact Craig through www.snubcommunications.com



Files on disc

You'll find the files you need to complete this tutorial on the cover disc.

FLOATING TECHNIQUES

1. The navigation and content DIVs (seen here in red and purple) are both floated left, so they sit next to each other, so creating columns in CSS.

2. The footer (shown in cyan) is set to clear all floated content. This means that, no matter what the height of either of the floated columns (or content within), the footer will always sit underneath.

3. The boxout (shown in yellow) sits within the content DIV and is floated right so that any other content wraps around it. Margins are controlled by CSS.

4. By using contextual selectors, you can define styles for elements within specific page areas. We've set borders within the boxout to have a three-pixel border, but this style won't affect the other images.

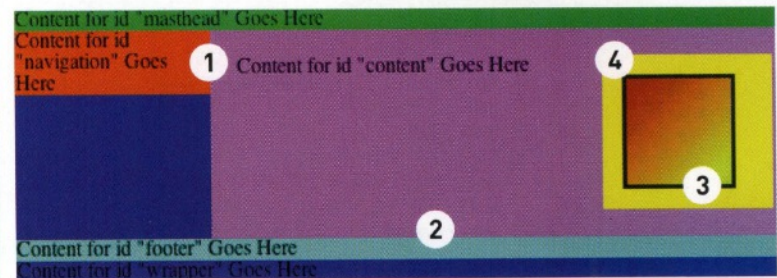


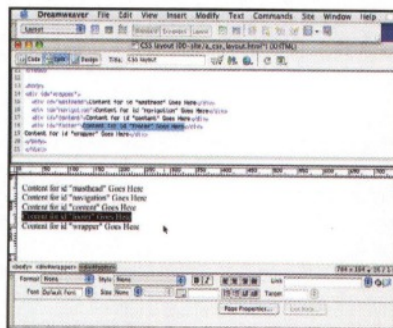
Illustration Kerry Roper

Part 1: Create a web page layout

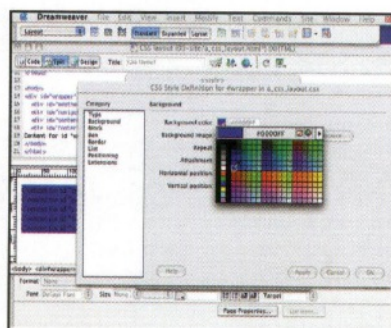
Follow these tips to craft your first CSS-friendly web page – and don't be put off by some of the arcane language. Once you get to grips with the process, it soon becomes second nature...

Box model

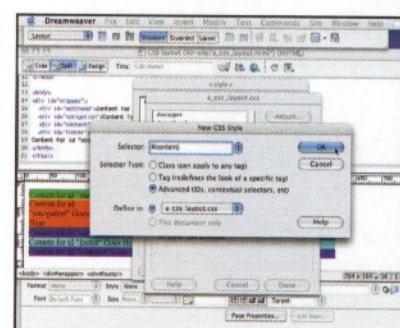
In the CSS box model, each element has a containing box, whose dimensions you can define. To this, you can add padding (to the edges of which backgrounds stretch), borders and margins. Internet Explorer 5.x for Windows gets the box model wrong, placing padding and borders inside the defined box dimensions rather than adding them to it. Therefore, when creating CSS layouts, you may need to use Tante's Çelik's box model hack (www.tante.com/CSS/Examples/boxmodelhack.html).



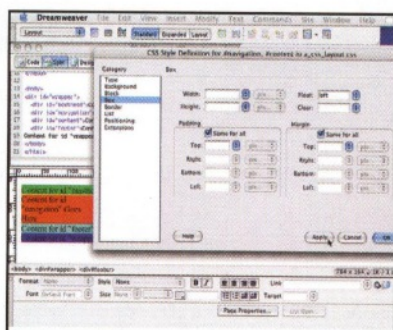
1 First, you need to define the page structure. Add a DIV with an ID of "wrapper". Nest four more DIVs within it with the IDs "masthead", "navigation", "content" and "footer" (in that order). It should look like the screenshot here – if not, you may need to rearrange the DIVs (which can easily be done in Code view).



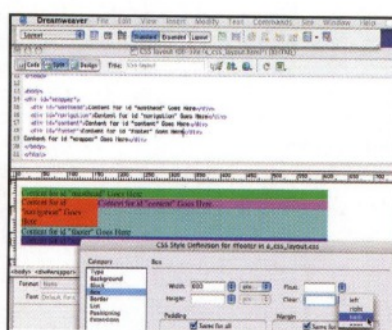
2 Now to style the wrapper. Click New in the Style dialogue to access the New CSS Style dialogue. Select Advanced and your CSS file for Define In. In the Selector field, type "#wrapper" (minus the quotes). In Background, set the background colour to blue and, in Box, set the Width to 600 pixels.



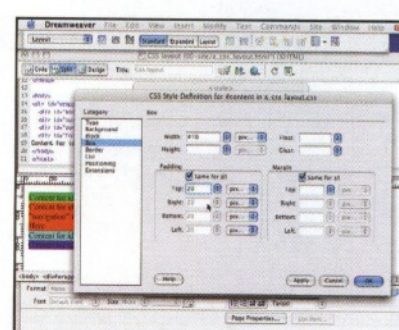
3 Style the remaining DIVs in the same manner, remembering the hash ("#") sign before each ID (as before), and assign a unique background colour to each. Set the Width of #masthead and #footer to 600 pixels, #navigation to 150 pixels, and #content to 450 pixels. You're now prepped for Step 4.



4 Time to set your floats. The DIVs currently appear one on top the other, but you need the content DIV to sit to the right of the navigation DIV, like two columns. This is where the float function comes in handy. Use the Advanced selector "#navigation, #content" (without quotes) and, in Box, set Float to Left.



5 Ready to clear your floated elements? From the coloured backgrounds above, you'll see that the footer content wraps around the last floated DIV (content). However, the footer needs to appear underneath all floated content. Do this by editing the #footer rule, and setting Clear (in the Box category) to Both.



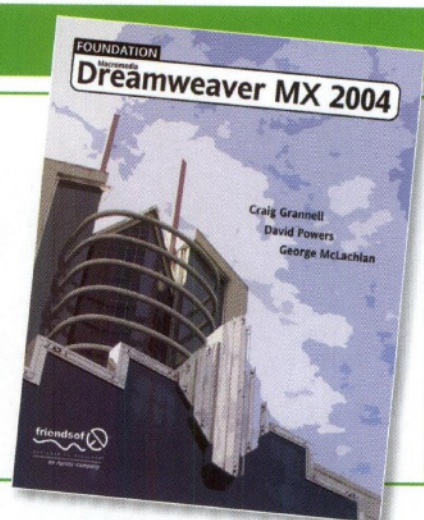
6 If you want to add padding to an area, the space this takes up must be removed from the set width (see "Box model" margin note, left). For instance, if you edit #content and add 20 pixels of padding, you'll need to subtract 40 pixels from the width, leaving 410 pixels. See, those maths classes proved useful, after all!

READ THIS...

Foundation Dreamweaver MX 2004
by Craig Grannell, David Powers and George McLachlan
PRICE: £22
PUBLISHER: Friends of ED
ISDN: 1-59059-308-1

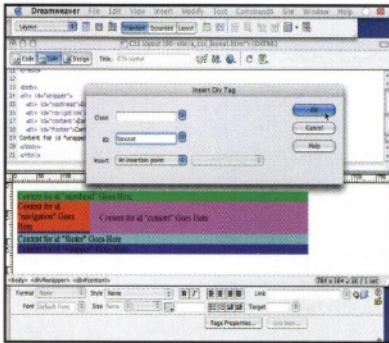
If you want to learn more about using Dreamweaver MX 2004 to create cutting-edge CSS-based layouts, check out Foundation Dreamweaver MX 2004, which provides a practical guide to Dreamweaver's key features, starting with basic layout tasks, working through the

creation of a static web site based on templates, and ending with a fully dynamic database-driven site. The first half concentrates on creating a site with a CSS-based layout using Dreamweaver MX 2004's Design view, CSS tools and templates. A "bridging" chapter then explains how to work directly with code, before leading you through the process of turning your static web site into something more dynamic, complete with content management system.

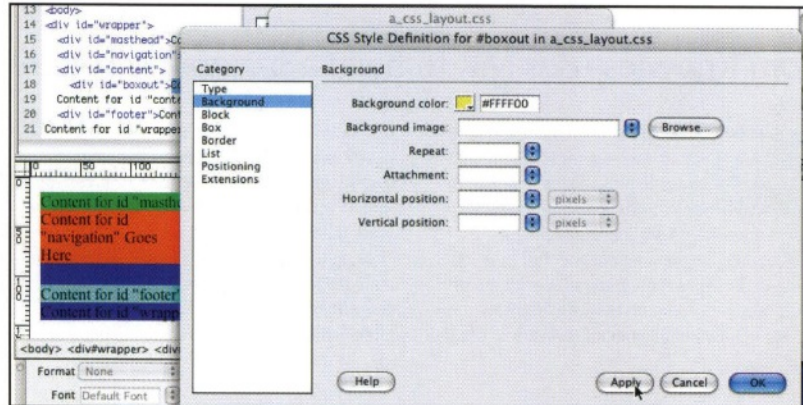


Part 2: Online floated elements

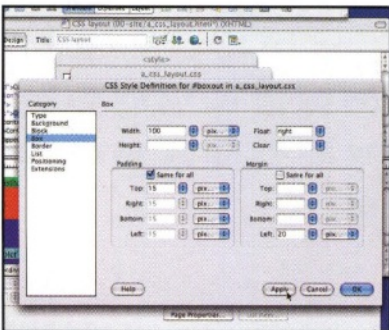
Once Float directions are assigned to elements in your web page, content automatically wraps around them, making updates easier to manage...



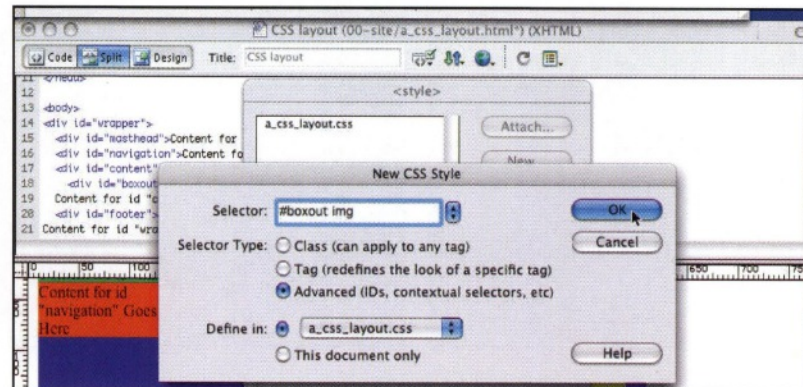
1 First, you need to create suitable elements for your web page. For this tutorial, you'll create a floated box into which you can place an image. Position the cursor in front of the element you want the floated element to float to the right of, then add a DIV with an ID of "boxout", as shown.



2 Now to create a new style. As in the first tutorial, choose Advanced in the New CSS Style dialog box. Enter "#boxout" for the style name and hit OK to access the CSS Style Definition dialog box. Set a unique background colour using the Background category.



3 In Box, set Width to 100 pixels, Float to Right, padding to 15 pixels, and Margin Left to 20 pixels (uncheck Same For All to set individual margin properties). Click OK until all dialogs are closed and then replace the default box out DIV content with an image.



4 In the New CSS Style dialog box, create a new style and call it "#boxout img". The property values defined for this style will only affect images within the boxout DIV. In Border, set the values to Solid, 3 pixels and #333333 to define a thick, solid border around images within the boxout, but not elsewhere.

CSS selectors

The selector is the part of a CSS rule that determines which element of a web page a style should be applied to. There are various types of selectors: element or tag selectors apply a style to an XHTML element, whereas contextual selectors apply styles based on context (such as "td p" for paragraphs within table cells). Class and ID selectors, which in the CSS file are preceded by a full stop and hash sign respectively, enable you to apply styles to elements with a specific class attribute value or a unique ID.

CSS declaration

The second part of a CSS rule is the declaration. This is enclosed in curly brackets and comprises one or more property/value pairs, separated by semi-colons (color: black; width: 50px). Various rules exist for the properties used. The www.w3schools.com site has a straightforward reference guide that's best used when you're coding directly, and Dreamweaver's Help panel has a CSS1 guidebook.

TOP TEN TIPS

1. Before creating CSS layouts, carefully plan out your page structure. Break your design down to its simplest elements, name each area, and then add them as DIVs with unique IDs, which can then be styled up.

2. When creating a page, check the "Make document XHTML compliant" box (bottom-right of the dialog) to ensure your page uses the current mark-up standard.

3. When attaching style sheets, select Import from the Add As options. This attaches CSS files in a way that hides them from obsolete browsers that have trouble with CSS.

Anyone using such browsers will still be able to access the content.

4. In the tutorials shown here, we used coloured backgrounds to differentiate each styled DIV. This can be useful when creating your own site, but can easily be removed once the layout is completed.

5. Although much improved over previous versions, Dreamweaver's Design view is a good guide, but not necessarily an accurate portrayal of your work. Always test pages in a range of browsers.

6. Don't immediately remove the default content Dreamweaver adds

for new DIVs. It will help you visualise your page structure and place new elements. Only delete it once you've nailed your page down.

7. Use semantic mark-up. In other words, use relevant tags – headings rather than styled paragraphs for headings, and XHTML lists rather than line breaks for list items. As well as maintaining a coherent document structure, semantic mark-up makes it easier to style elements in CSS.

8. The World Wide Web Consortium (W3C) offers a number of validation services on its web site at www.w3.org. Use them to validate your

templates and CSS files and weed out errors or compatibility issues.

9. Make an effort to understand how XHTML and CSS work, and you'll be able to work faster, creating rules directly. You'll also be better equipped to fix errors.

10. Learn how to use XHTML and CSS comments to isolate areas of code, or to temporarily hide objects. This will help you locate problems and squash any bugs.



Studio MX: Virtual tours

Tom Knowles uses MX to build a panoramic picture viewer

This might rub you up the wrong way if you're a fervent anti-capitalist, but Macromedia looks to be on a quest for world domination with its Flash player.

A couple of years ago, it seemed that you needed a different plug-in for every type of multimedia content on the web. Want to play games? Download Shockwave. Want to chat?

Download Java. Want to watch video? Download RealPlayer. Oh, and download QuickTime and Windows Media Player, too. And, if you want to do any of this on a device other than a PC, forget it.

Now, the Flash player can pretty much do it all. And it's been providing Flash developers with the capabilities to build effective games and powerful online

applications, and essentially negate the need for so many different plug-ins. Macromedia has also developed Flash's cross-platform capability so you can now get the player on your mobile phone.

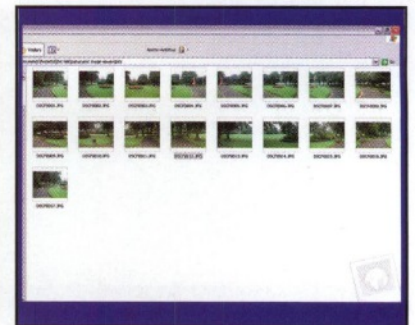
We're going to demonstrate this jack-of-all-trades by creating a panoramic virtual tour — the sort that used to be the sole domain of a variety of obscure plug-ins. ■

Part 1: Put your picture together with Fireworks

First head off to the wide, open spaces of your local park and take your photograph...

Pick a preloader

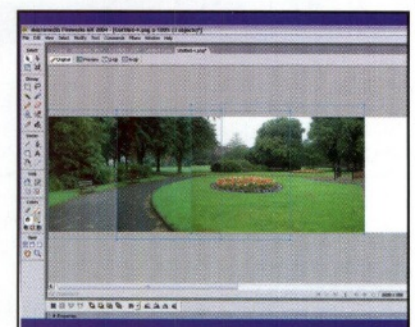
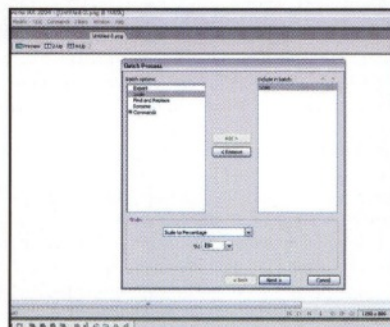
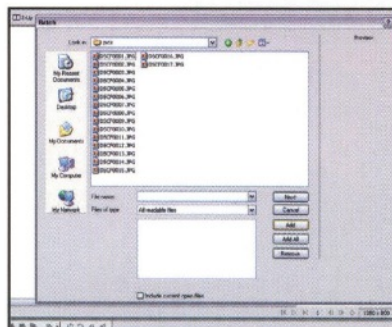
If you're planning on using large images (both in terms of bytes and pixels), be sure to include a preloader (there's one in the components panel that you can drop into your movie).



1 To create a panoramic image, the first thing you'll need to do is to take your camera (preferably a digital one) to wherever you want to showcase and take some pictures. Ideally, your location needs to be quite large if you want the effect to be as dramatic as possible.

2 For this project, I chose my local park as my location. To get the panorama to be as realistic as possible, you'll need to be able to hold or fix your camera at a consistent height and angle, and then turn through 360 degrees, taking a picture roughly every 20 degrees.

3 Transfer your shots to your hard drive, you'll need to reduce their size, ideally to 400 to 500 pixels in height. The percentage reduction obviously being dependent on their original size. Once you know what sort of percentage you want to reduce them by, open Fireworks MX 2004.



4 Rather than having to resize each individual image by hand, Fireworks provides us with a rather handy batch process wizard. Choose File > Batch Process, and when the prompt appears, pick out all your raw photos, clicking Add. When you've selected them all, click Next.

5 In the Batch Options panel, choose Scale, and then hit Add. In the Scale dropdown, you'll be able to specify the percentage we worked out earlier. Hit Next again, and lastly you'll be able to specify the location for your modified files. Hit Batch to run the process.

6 When all your images have been resized, choose File > New and set a document size of around 2000 to 3000 pixels by 300 pixels. Once you've done this, start opening, copying and pasting each resized photograph into the document, taking care to nudge each one along.



Part 1: Put your picture together with Fireworks

continued



7 Now you've got all your images in one big row, it's time to start lining them up. This can take some time, and requires some careful positioning and a couple of sneaky tricks. The first problem you'll encounter will be changes in the brightness of adjacent images.



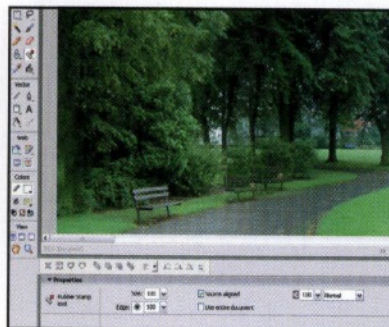
8 This is where the good old Eraser tool comes into play and, if you've taken enough photos to mean that there is plenty of overlap, this technique can work wonders. In the Eraser properties panel, ensure the size and edge are set to 100 and the opacity is 50.



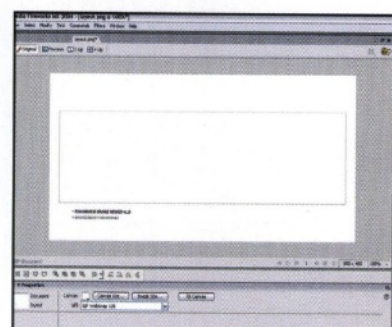
9 If there is a visible seam in your images, select either the underlying or overlying image, and carefully erase the edge, blending the two images together. If there's a difference in the brightness, use the marquee tool to select the area and brighten it with the Filters > Adjust Colour panel.



10 If you were careful not to change your shot angle when you were taking your photos, lining them all up along the document's length should be reasonably straightforward. The only tricky part comes when you want to make both ends match up seamlessly. This sometimes requires some cheating.



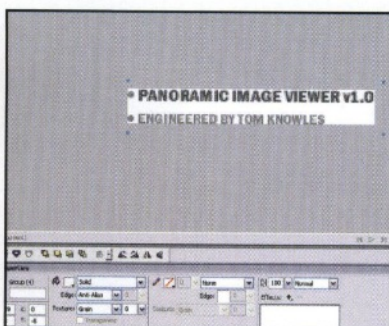
11 By cheating, I mean making use of the clone stamp to patch samples of the left-most image onto the right-most image. The best way of checking your progress is to import the PNG into Flash, place three instances of it side by side, and then update it whenever you've made changes.



12 By constantly tweaking the size of the document and smudging the edges, you should eventually have a seamless montage image. Now create a new document in Fireworks, of 800x400 pixels, and draw a similar layout as the one above. Place a white rectangle of 744x213 pixels in the centre.



13 Underneath the rectangle, paste the entire montage and then choose Modify > Mask > Group as Mask. To create the custom cursor, place a filled circle 4 pixels in diameter inside an outlined circle 10 pixels in diameter, and then use the Polygon tool to draw a 4x4 pixel triangle.



14 With our simple interface design complete, all we need to do is select the credit details we drew and paste them into a new document. Once done, choose Modify > Canvas > Trim Canvas, and then File > Export Preview, saving your image as a GIF.

Go further

Just because our application simply allows you to scroll a picture, doesn't mean yours should stop there. Why not base your web site's actual navigation around our panoramic engine?

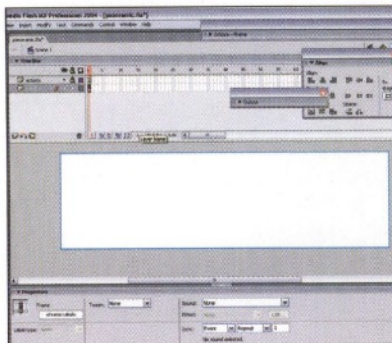


Part 2: Create the viewer with Flash

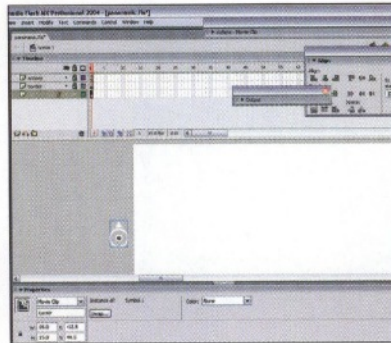
Now it's time to place our code into Flash...

Final test

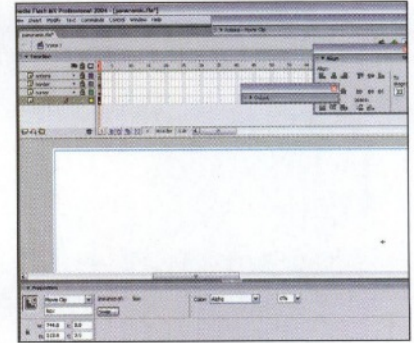
To test the finished movie out within the authoring environment, choose File > Publish Preview > Flash.



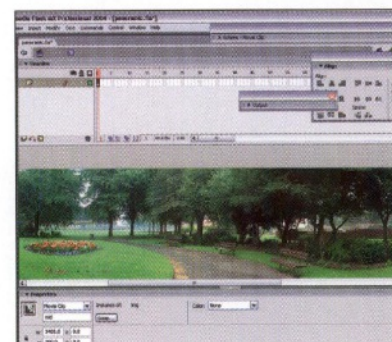
1 Open Flash, and create a new document of 750x220 pixels, with a frame rate of 81 FPS. Rename the first layer to Actions – this is where we'll place our code. Then create another new layer called Border. On this layer, draw a grey-outlined rectangle the same size as the stage.



2 Below, create another new layer, and paste the cursor image from our Fireworks file onto the Flash stage. Choose Modify > Convert to Symbol > MovieClip, ensuring its circle is centred within its stage, and the arrow is facing upwards. On the timeline, give it the instance name cursor.



3 On the next layer, create another rectangle, this time of size 744x213 pixels. Convert it to a MovieClip, and give it the instance name box, and use the Align panel to centre it on its own stage and the main timeline. In the properties panel, set its alpha value to 0.



4 Create another layer, and choose File > Import > Import to Stage, and select the panoramic image file we created earlier. Once on stage, convert it to a MovieClip, and give it the instance name mid. Now place it inside another parent MovieClip, with the instance name imgs.

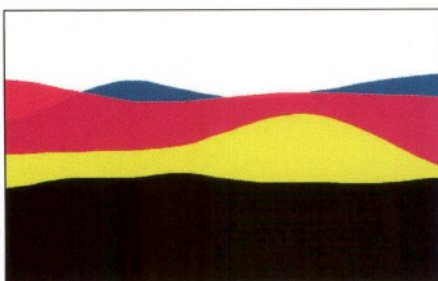


5 On the main timeline, on a layer above the imgs MovieClip, draw a rectangle of width 744x213 pixels. Use the Align panel to centre both it and imgs on the stage, and then select its layer. Right-click and choose Mask. You should now have a neatly masked window into the image.



6 Once you've added the following ActionScript code, our Flash file is complete. Choose File > Publish Settings and make sure you're publishing in ActionScript 1.0 for the Flash 7 Player.

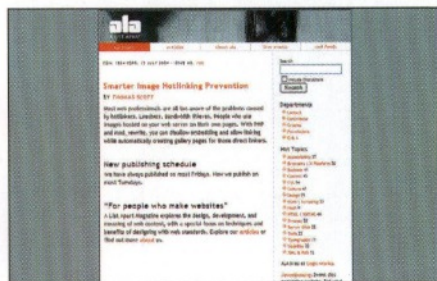
Resources: Inspiration, perspiration and motivation



Yugo Nakamura

Yugo's revamped web site is an utterly astonishing exploration of ActionScript-driven interface design. Take a look, it's inspiring stuff.

www.yugop.com



A List Apart

As we steadily move away from traditional table-based HTML layouts, A List Apart is easily the best place to go for the latest CSS techniques.

www.alistapart.com



Jeffrey Zeldman

CSS expert Jeffrey Zeldman's personal site contains lots of great links and articles about the move towards XHTML and standards compliance.

www.zeldman.com

Part 3: The trigonometry of Flash

Flash provides plenty of trig methods, none of which require a calculator...

At the time of Flash 5, Yugo Nakamura (see resource 1) showcased panoramic sliding interfaces and inertia-based mouse followers, which blew everyone's minds with the sheer amount of code involved. With Flash MX 2004 ActionScript we can achieve the same effects with a fraction of the code. We can forget about implementing such mundane functionality and focus on building on top of it.

1 Lines 1–2: First up, we need to import some external ActionScript files (included on the CD). As regular readers will know, the first of these files consists of Robert Penner's easing equations, and the second contains some custom movement prototype methods for the MovieClip

2 Lines 3–6: Here, we utilise the Stage object to remove elements of the right-click menu, and also hide the mouse cursor. The XTopSpeed and YTopSpeed variables represent the maximum speed at which we can move our panoramic images. Finally, we set the initial X and Y speeds of the image as zero.

```
Stage.scaleMode = "noScale";
Stage.showMenu = false;
Mouse.hide();
XTopSpeed = 10; YTopSpeed = 2;
stage_width = 750; stage_height = 220;
imgs.Xspeed = 0; imgs.Yspeed = 0;
```

3 Lines 7–13: In order to give the impression of continuous movement, we need a spot of trickery to create two duplicates of our main image MovieClip, (imgs.mid), and then place them behind and in front of it. Notice how we use the imgwidth variable in order prevent us from having to hardcode the width of the image.

```
imgwidth = imgs.width;
duplicateMovieClip("imgs.mid", "back", 1);
imgs.mid._x = -imgwidth;
duplicateMovieClip("imgs.mid", "front", 2);
imgs.mid._x = imgwidth;
backlimit = -imgwidth;
ydiff = imgs.height - stage_height;
```

4 Lines 14–26: Now we create an onEnterFrame function on the imgs MovieClip. Every time this function is executed, it moves the image according to the current speeds. It also, however, examines the overall position of the MovieClip. If the position is over the limit, the function resets it and creates the illusion that the image is continually scrolling.

```
imgs.onEnterFrame = function(){
    this._x -= this.Xspeed;
    if(this._x <= backlimit){
        this._x = 0;
    }
    else if(this._x >= 0){
        this._x = backlimit;
    }
    futureY = this._y - this.Yspeed;
```

```
if(futureY <= 0 && futureY >= 0-ydiff){
    this._y -= this.Yspeed;
}
```

5 Lines 27–51: In this segment, we use one of our movement prototype methods to move the custom cursor towards the mouse every time the mouse is moved. In addition, we perform some basic trigonometry that uses the principles of right-hand triangles and tangents to find out where the mouse is in relation to the custom cursor, and then rotates the custom cursor to point in that direction.

```
root.onMouseMove = function(){
    cursor.easeOutMove(_root._xmouse, _root._ymouse, 40);
    ydist = _root._ymouse - cursor._y;
    xdist = _root._xmouse - cursor._x;
    yval = ydist; xval = xdist;
    if(ydist < 0){
        yval = -1*ydist;
    }
    if(xdist < 0){
        xval = -1*xdist;
    }
    degrees = Math.atan2(yval,xval) / (Math.PI/180);
    angle = 90 - degrees;
    if(ydist > 0 && xdist > 0){
        cursor._rotation = 180 - angle;
    }
}
```

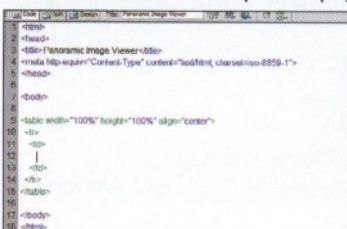
```
else if(ydist < 0 && xdist > 0){
    cursor._rotation = 0 + angle;
}
else if(ydist < 0 && xdist < 0){
    cursor._rotation = 0 - angle;
}
else if(ydist > 0 && xdist < 0){
    cursor._rotation = 180 + angle;
}
```

6 Lines 52–57: Finally, we do some quick calculations to work out where the mouse is in relation to the centre of the stage. Then we can work out the imgs MovieClip's Xspeed and Yspeed variables and hence how fast the image should be moving.

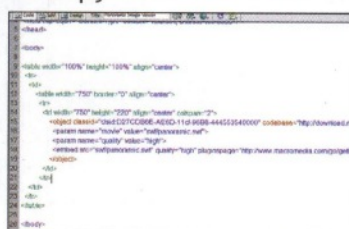
```
Xlength = stage_width/2;
Ylength = stage_height/2;
xval = box._xmouse; yval = box._ymouse;
Xpercent = xval/Xlength;
Ypercent = yval/Ylength;
imgs.Xspeed = XTopSpeed*Xpercent;
imgs.Yspeed = YTopSpeed*Ypercent;
```

Part 4: Putting the panoramic shot in your viewer

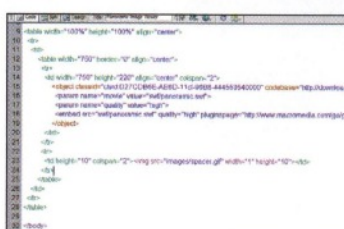
Now it's time to open up your copy of Dreamweaver...



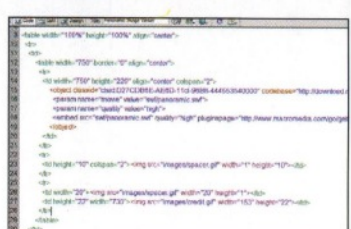
1 In Dreamweaver choose Create New HTML. Switch to Code View, and type in a document title. Create a 100%x100% table. For this to display properly in all browsers, remove the doctype tag.



2 Add a nested table in the first table, with a width of 750. Give the first cell a width and height of 750 by 220 pixels, and specify the colspan as 2. Choose Insert > Media > Flash and choose our panoramic.swf file.



3 Create another row, with two cells. Insert a 20 pixel-wide spacer GIF into the first cell, and the credit image into the second. And that's it – save your HTML file and then view the end result in any browser.



4 Add another row in the nested table, with a height of 10 and a colspan of 2. Insert the transparent spacer GIF (from the CD), giving it a height of 10. This forces white space between the Flash file and our credit image.

Studio MX: Photo album

All the components needed to complete this tutorial can be found on the cover disc.

Tom Knowles show us how to use Macromedia Studio MX to design and build a Flash-based photo album...

With the increasing ubiquity of digital cameras and the explosion in blogging activity, more people are publishing their photos on the web. Whether they're wedding pictures or just evidence of a drunken get together, publishing your photos online is a great way to showcase and store them. Using Studio MX, we can create a smooth dynamic Flash interface to display

your photos and a simple admin page to upload them with.

Rudimentary Flash albums are fairly commonplace, of course, but one of the frequent drawbacks is that they rely on hand coded XML data sources (which are tedious to update), and they usually require all images to be the same size.

However, using some clever ActionScript we can accomplish some sexy things like

fading the photos in and out, and actually resizing the interface to fit.

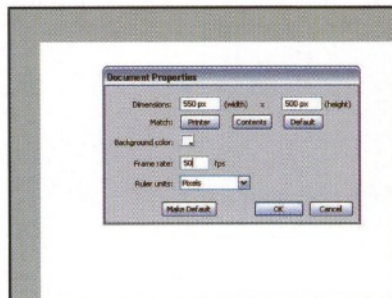
Here we're concentrating on ActionScript and setting up the Flash interface, then we'll tweak the ASP with Dreamweaver. The sample ASP files in this tutorial come courtesy of top-notch ASP.NET developer John Jobling (www.travellerstales.tv). ■

Part 1: Using Studio MX to build the photo album

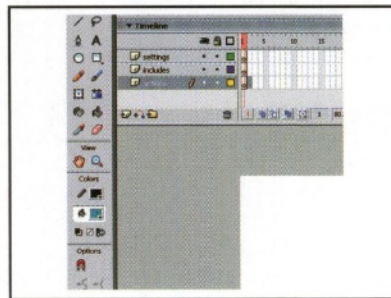
First, we'll create the Flash interface elements and organise our movie...

Password protect

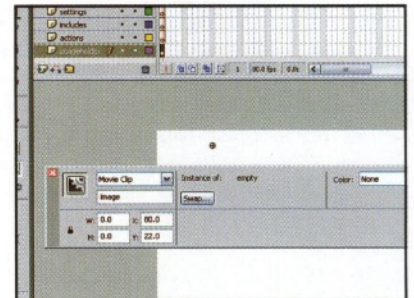
To prevent unauthorized users uploading pictures to your album, why not visit one of the ASP developer sites to learn how to password-protect your upload.aspx page?



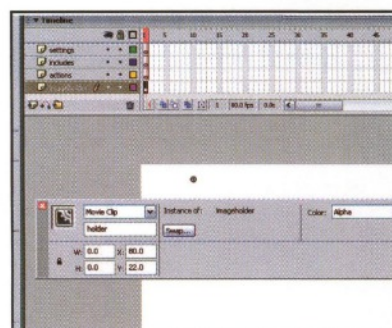
1 First, we're going to create the Flash interface elements and organise our movie. Open MX, click New Flash Document and choose Modify > Document. Set the dimensions to 500x500 pixels and a framerate of 80fps. We're using a high framerate to ensure smooth transitions.



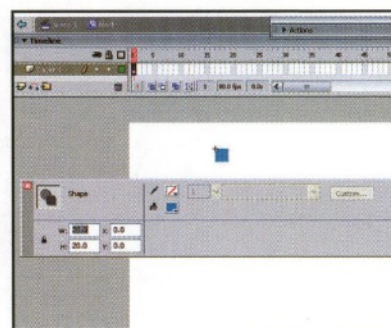
2 Double-click the first layer on the timeline, and rename it 'settings'. We're going to place some code on this layer later on, so leave it blank for now. Add two more blank layers, named 'includes' and 'actions'. Again, we'll return to these later, so leave them both blank.



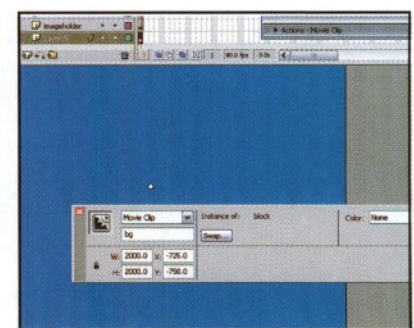
3 Next, create a blank Movie Clip called 'empty', and drop it onto a new layer called imageholder, anywhere on the main timeline. Give it the instance name 'image'. With it selected, choose Modify > Convert to Symbol > MovieClip, giving this new parent movieclip the name imageholder.



4 Give this parent clip the instance name 'holder', and in the properties panel, set its alpha value to 0. This MovieClip is the clip which we'll be fading in and out, and its child image MovieClip is what we're going to actually load our JPEG images into.



5 Create another new layer and then use the rectangle tool to draw out a square of size 20x20 pixels. Choose Modify > Convert To Symbol > MovieClip and give it the name 'block'. We're going to use this block clip for a number of purposes.

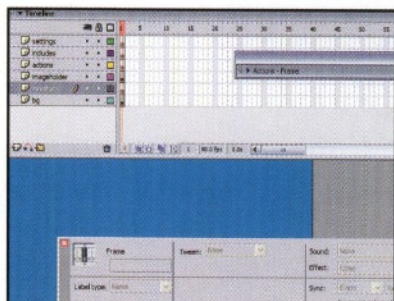


6 Back on the main timeline, select the instance of the block MovieClip we just created, and give it the instance name 'bg'. In the properties panel, give it the dimensions of 2000x2000 pixels, and then in the align panel, centre it relative to the stage.

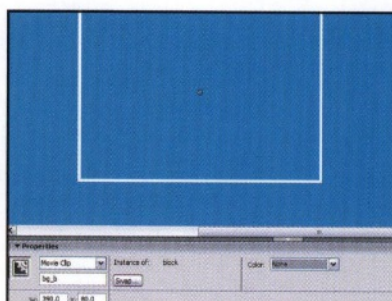


Part 1: Using Studio MX to build the photo album

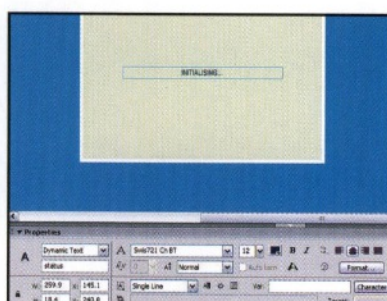
continued



7 The reason for this huge block background is that we can use it as a reference for keeping the main interface constantly centred in the browser, even when we are resizing to fit the loaded picture. Next, create another new layer above the bg MovieClip, and label it 'interface'.



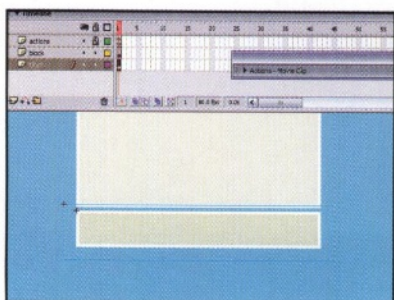
8 Drag a new instance of the block MovieClip from the library onto the stage, and in the properties panel, change its tint colour to white, give it the instance name 'bg_w' and give it dimensions of 400x290 pixels. Now overlay another block clip and give it the name 'bg_b'.



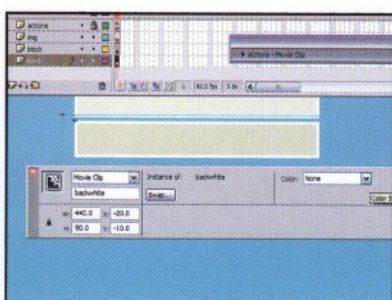
9 Change the bg_b's tint to a beige colour as above, and give it dimensions of 390x280, ensuring that it and bg_w are centred. In addition, overlay a dynamic textfield with the instance name 'status'. Now choose Insert > New Symbol > MovieClip, and create a clip called 'Nav'.

Scrollbar

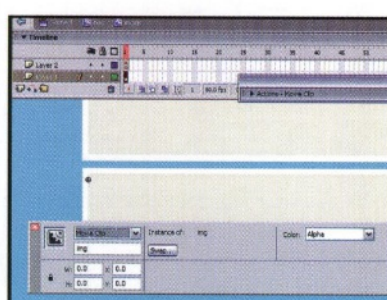
In our version of the photo album, the number of image thumbnails displayed is limited to 8. To combat this, why not use the scrollbar component?



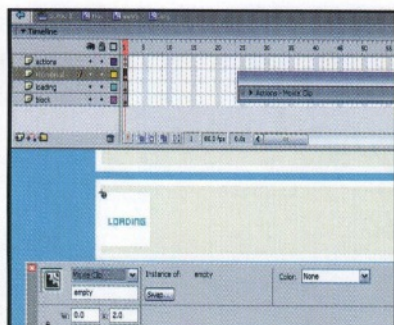
10 Create a new layer, and place another beige-tinted instance of the block MovieClip, with dimensions of 390x50 pixels and the instance name 'Back'. Add another layer underneath this one, and place a 400x60 block, in white, over a transparent rectangle of 440x90 pixels, as above.



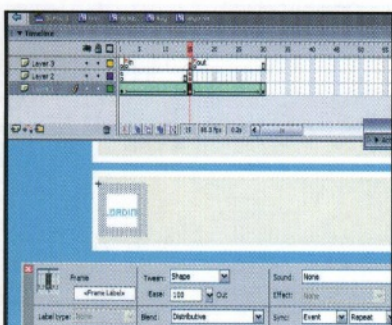
11 Group the white and transparent blocks and convert them to a MovieClip called 'backwhite' with the instance name 'backwhite'. You should now have a setup that matches that of the bg_w and bg_b clips on the main timeline. Create one last layer above these blocks called 'img'.



12 Choose Insert > New Symbol, and create a MovieClip called 'imgs'. Inside imgs, add a simple stop() action and then create another layer. Here, insert another new MovieClip called 'img', and give it the instance name 'img'. This is the MovieClip we'll be duplicating later, so drill down to edit it.



13 Place a 40x40 white block called 'block'. Above that, place another block, without any instance name, with dimensions of 36x36 pixels. On a layer above, create a MovieClip called 'loadingtxt' with some suitable text. Above that, place an instance of empty, called 'empty' at coordinates of 2,2.



14 Create a layer above the rest and a new MovieClip with the instance name 'imgover'. Inside this clip, create a mouseover animation, with frames labelled in and out to indicate the start of the rollover and rollout sections. Put stop() actions in the places where you want the animation to stop. ➤

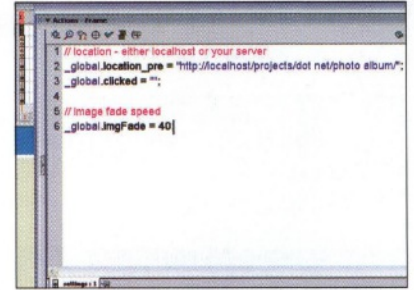
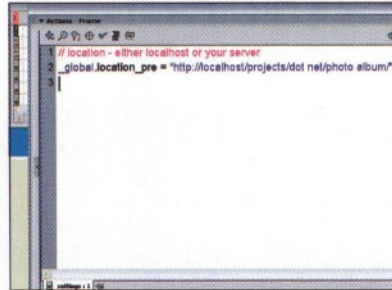
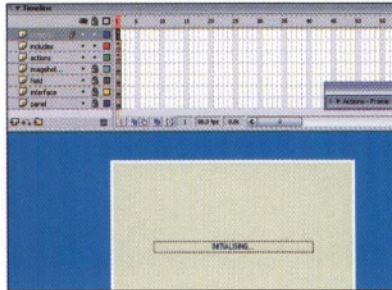


Part 2: Making your album flexible

Before we look at the bulk of our ActionScript, we need to add some code...

Variable fading

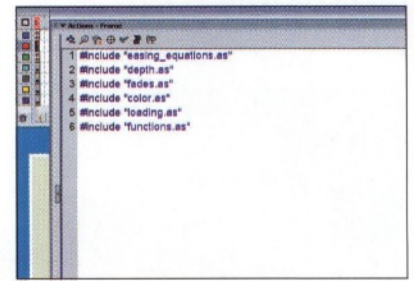
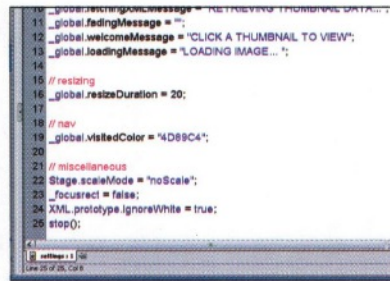
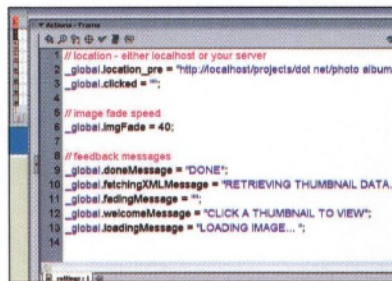
The variable fading message is displayed while the images are being faded out.



1 Since we want our photo album to be as flexible as possible, it makes sense to create a settings layer, which allows us to easily configure the movie without delving into the complex code.

2 On the settings layer, define global variables used by the functions to implement their functionality. First, we define a variable called 'location_pre'. This represents the album's http address. If you test locally, with IIS or PWS, this must be http://localhost/photoalbum or similar.

3 The clicked variable initialises the placeholder we use in our code to remember what has been clicked. The next variable is imgFade. This represents the number of frames that the script takes to fade the large image in and out. This will obviously be affected by the movie's framerate.

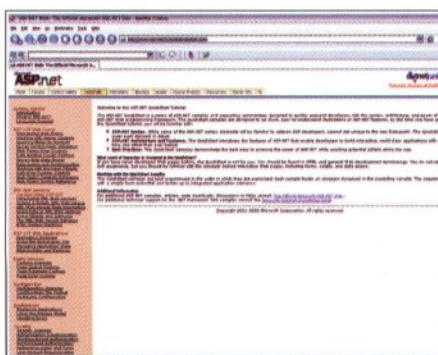


4 Next, we have some feedback messages, which will be displayed in the status textfield that overlays the interface. These are self-explanatory, and are displayed during parts of the scripts procedures.

5 resizeDuration and visitedColor are the frames used by the interface to resize, and colour of a visited thumbnail border, respectively. We've also added settings, such as preventing the movie being scaled; stopping tab button usage and ensuring XML whitespace is ignored.

6 On the includes layer, we need to include some external ActionScript files. Amongst these are some of Robert Penner's easing equations and the code to handle the resizing. Lastly, in Publish Settings, make sure to set the load order as top down, and that we're using ActionScript 1.0.

Resources: check out these sites to advance your web design skills



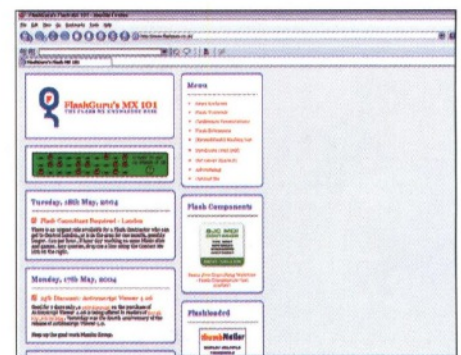
Microsoft ASP.NET

The place to start if you want to learn the nitty-gritty of ASP.NET programming.
www.asp.net/Tutorials/quickstart.aspx



ASP Free

Another ASP resource, this one is independent and carries some excellent tutorials and samples.
www.aspfree.com



FlashGuru

Guy Watson's site is an great resource for intermediate to advanced developers wanting to enhance their knowledge.
www.flashguru.co.uk

Part 3: Coding the layout of the album

XML, MovieClipLoader and button actions...

In the following code, we'll demonstrate how to retrieve and process the XML feed containing the thumbnail images, create a MovieClipLoader and call and assign some button actions to the navigation. Due to the complexity of some of the resizing code and the length of the scripts required to initialise the MovieClipLoader class, we've moved that functionality to external the ActionScript files, which are included on your cover disc.

1 Lines 1–4: First, we need to retrieve our XML feed from the xml.aspx file.

This file returns a list of all the images in the uploads folder, complete with references to their associated thumbnails. Notice how we supply the onLoad handler with the function processImages before we initialise the load call. We also place our predefined fetching XML Message in the status textfield.

```
imageXML = new XML();
imageXML.onLoad = processImages;
status.text =
fetchingXMLMessage;
imageXML.load(location_pre+
"xml.aspx");
```

2 Lines 5–12: The beginning of the processImages function. Rather than constantly having to dig into the imageXML object's structure, we simply stick the branch we're interested in into the variable thumbs. We then create the classic Duplicate MovieClip For Loop, duplicating the img clip currently sitting in the nav MovieClip. Before we load the actual thumbnail image, we ensure the user can't click the clip by setting its enabled property to false.

```
function processImages(){
var xval = 5;
var thumbs = _root.imageXML.
firstChild.childNodes;
for(var i=0; i<8; i++){
duplicateMovieClip("nav.imgs.
img", "img"+i, i);
var clip = eval("nav.
imgs.img"+i);
clip._x = xval;
clip.enabled = false;
```

3 Lines 13–20: Okay, let's set up a MovieClipLoader object for the individual thumbnail. As well as instantiating a MovieClipLoader, we'll create a listener object to handle its events. Here, we're only interested in its onLoadComplete event. When that event is triggered, we make the loadingtxt MovieClip invisible and we enable the img clip itself.

```
clip.myLoader = new
MovieClipLoader();
clip.myListener = new
Object();
clip.myLoader.addListener
(clip.myListener);
clip.myListener.
onLoadComplete =
function(target_mc){
target_mc._parent.
loadingtxt._visible = 0;
target_mc._parent.
enabled = true;
}
clip.myLoader.loadClip
(location_pre+"thumbs[i].
attributes.thumb, clip.empty);
```

4 Lines 21–31: Here we fish out the path to the main image from the thumbs array and place it inside the clip's mainimage variable. We then deal with the rollover and rollout functions for the clip. In each case, we first check to see whether the clip has already been clicked by comparing it with the global reference clicked. If not, we fire off the mouseover animation.

```
clip.mainimage = thumbs[i].
attributes.path;
clip.onRollOver = clip.
onDragOver = function(){
if(this != clicked){
this.imgover.
gotoAndPlay("in");
}
}
clip.onRollOut = clip.
onDragOut = function(){
if(this != clicked){
this.imgover.
gotoAndPlay("out");
}
}
```

5 Lines 32–38: Next, let's write the onRelease statement for the clip. The first thing we do is remove the clicked status from the previously clicked MovieClip, if it exists. Once we've done that, we assign the new clicked value and disable the current clip. Notice how we also change the colour of the clip's background in order to indicate that it has been visited.

```
clip.onRelease = clip.
onReleaseOutside =
function(){
```

```
clicked.imgover.
gotoAndPlay("out");
clicked.enabled = true;
clicked = this;
clicked.enabled = false;
this.block.
setColor(visitedColor);
_root.temp = this;
```

6 Lines 39–50: The final task is to set up the onComplete handler which is fired off when the main image holder MovieClip has successfully been faded out (we fade the clip with the custom fade method which is included in the fades.as file). Once this has happened, we send out a new call to load the selected image into the _root.imgLoader object, and let our listener object handle the rest.

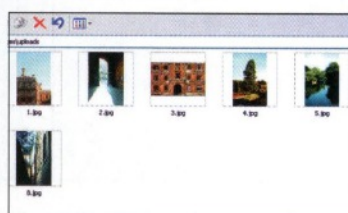
```
status.text =
fadingMessage;
_root.holder.onComplete
= function(){
status.text =
loadingMessage;
_root.imgLoader.loadClip
(location_pre+"_root.
temp.mainimage, _root.
holder.image);
}
_root.holder.
fade(0, imgFade);
}
xval += (nav.imgs.img._
width + 8);
}
status.text = welcomeMessage;
nav.imgs.img.killMovieClip();
}
```

Part 4: Uploading and viewing your album

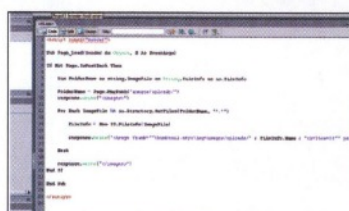
Everything needs to be in its correct folder...



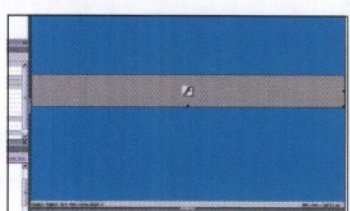
1 Our album's backend is based on ASP.NET, so if you're planning to test it on your own server, eg IIS, you'll need to download the Microsoft .Net Framework from www.asp.net. Wherever you host the album must run ASP.NET too.



2 All the .aspx files (included on the cover disc) must go in the root folder of your application. You should also have a folder called images. Inside images must be a folder called uploads; this is where the upload.aspx file will upload images to by default.



3 To configure your application differently, you'll need to alter two .aspx files. The files xml.aspx and upload_action.aspx have references to the where the uploaded files are stored, and are the files to delve into if you want to make changes.



4 Now we need default.aspx, containing the Flash file. In Dreamweaver, just drag and drop the Flash movie from the Files panel and set its dimensions to 100%x100%. The upload.aspx file is already set up so, to upload images, view it in a browser.



Studio MX: Flash guestbook



All the components needed to complete this tutorial can be found on the cover disc.

Tom Knowles uses Macromedia Studio MX to design and build a Flash-based guestbook...

Despite being around for years, the web site guestbook is still hugely popular. (Maybe it's because they've been around for years.) When Flash started to become more powerful, a handful of developers produced some full-Flash guestbooks and wowed the web design community. It seemed that these guys were using some sort of fendish necromancy to achieve such a

seemingly difficult goal. In fact, it's not that difficult and in this second tutorial we'll use Studio MX to design and implement a guestbook application, complete with a Flash interface and Fireworks graphics.

There are plenty of sample guestbook applications on the web and tying them into a Flash interface is simply a matter of making good use of ActionScript's external data retrieval capabilities. Many people struggle

with some of these, such as the LoadVars object, but they are deceptively simple, especially once you've got to grips with what they actually do.

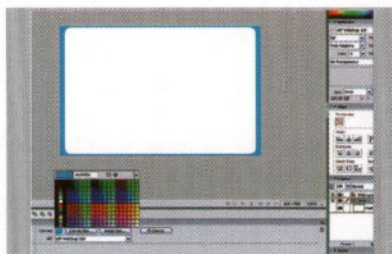
For this tutorial, we'll be using free guestbook ASP pages, available on Flashkit.com and written with Flash in mind. Our application will also use an Access database to store the entries. ■

Part 1: Laying the foundations

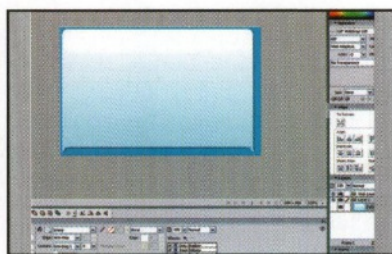
Building the basics of a guestbook in Fireworks

Data loading

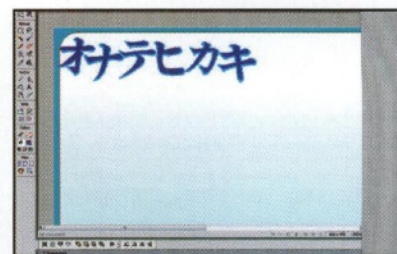
To provide more feedback on data loading for users on slow connections, make use of the LoadVars object's `getBytesLoaded()` and `getBytesTotal()` methods.



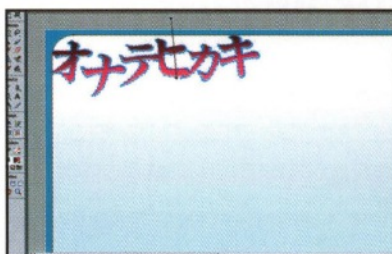
1 Open Fireworks and create a new document of 600x400 pixels. Give your background a suitable colour (in the document Properties panel), then select the rectangle tool. Draw a rectangle filling the canvas and then in the Properties panel set the roundness value to around 10.



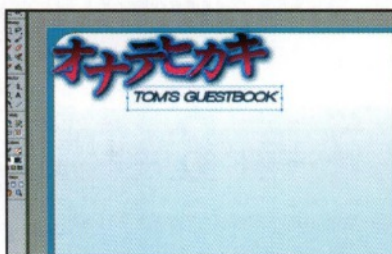
2 To create the inverted edges in our example, choose the sub-selection tool, and select the two curve points as above. Move them down with arrow keys until you're happy with the new curve. Now change the solid fill to a subtle gradient, and add a dropshadow in the Effects panel.



3 In the top right of our example, we've used some Japanese characters. Fonts like this are all over the web; just use Google to bring up a list of links and remember to check for copyright notices. Once you've typed out your characters, choose Text > Convert to Paths, and then Modify > Ungroup.



4 To enhance the cartoon-ish feel, jiggle the positions of the letters, using shift and the up and down arrows to change stacking order. When you're done, select all the characters and choose Modify > Combine Paths > Union. You can now apply a gradient fill to all the letters at once.



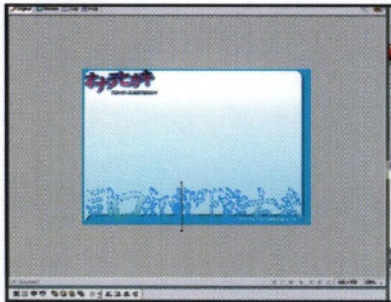
5 For a finishing touch, add a small dropshadow and a rounded bevel to your text in the Effects panel. As above, use the Type tool to type out an additional readable title in a font of your choice. Staying with the Type tool, write out a copyright notice in the bottom corner.



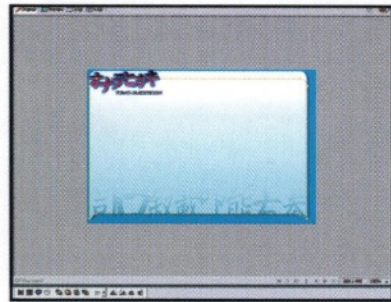
6 Next, we're going to create a large, masked set of Japanese characters along the bottom of the panel. First, type out a selection of characters in a large font. As before, choose Text > Convert To Paths, ungroup them and Modify > Combine Paths > Union.

Part 1: Laying the foundations

continued



7 Using the gradient fill, create a transparent to opaque fill as above. Then select our base panel, and copy and paste it over the top of our characters. Remove the dropshadow, and change the fill colour to solid white.



8 Now select both the characters and the white shape, and choose **Modify > Mask > Group As Mask**. As above, you'll see that the characters are cropped to the shape of the underlying panel. Masking in this manner is a useful technique, and is worth experimenting with.



9 Make another copy of our base shape, and paste it underneath the Japanese characters and title we drew out earlier. Modify the colour and, as above, use the sub-selection tool to select the four points at the bottom and decrease the overall size, forming a header element.

For ASP whiz-kids

If you fancy yourself as an ASP whiz, why not dip into the `loadvalues.asp` file and reconfigure it to deliver more than one of the guestbook entries at a time?



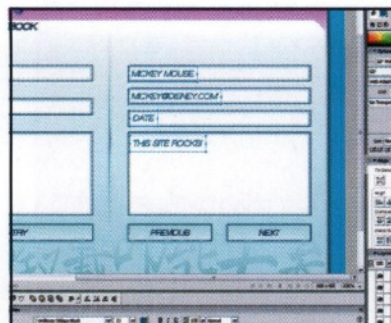
10 To create the crease in the centre of the panel, draw a rectangle and in the Properties panel, set its width to one pixel. Change its fill properties to a triple gradient, with the outermost colours being transparent. To finish, add a light, one-pixel dropshadow.



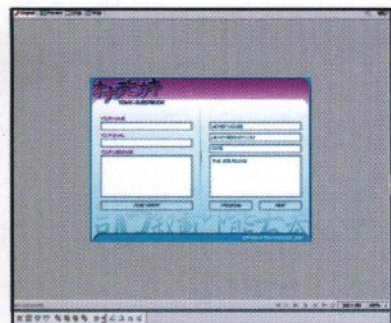
11 Drawing the guestbook entry fields is a simple matter: choose the rectangle tool, and select a white fill and a one pixel outline. Add a soft inner glow in the effects panel to give us an extra feeling of depth. To keep the shape crisp, ensure the edge is hard.



12 Use the Type tool to add labels to the fields. The buttons are very similar to the fields - just remove the inner glow, and replace it with a drop shadow. As above, change the flat fill to a vertical gradient, and add your button labels with the type tool.



13 It's useful when producing a mockup to spend time on the details of your layout. Since we're going to use our file directly within Flash, you'll need to remove any text that could conceptually be changed by the user.



14 Before we save our graphic and import it into Flash, choose **Modify > Canvas > Trim Canvas**. This will eliminate any unnecessary space from around the graphic, cutting down on precious file size. Once you've done that, save your file as a PNG in the normal manner.

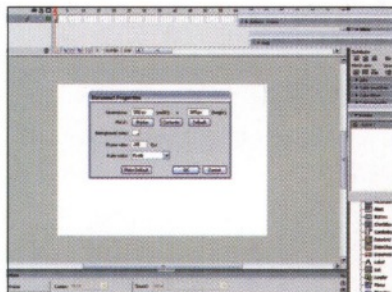


Part 2: Making our buttons respond

Animating our guestbook in Flash...

Size doesn't matter

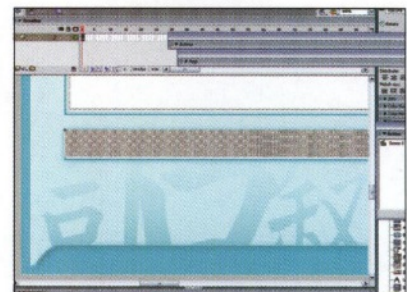
Your movie clip can be of any size – we'll resize it once we've created some instances of it.



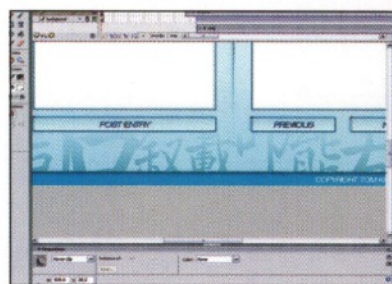
1 Open Flash and create a new document. The frame rate and background colour are unimportant since we're concerned with the scripting and we've already created our interface layout. Set the stage size to the same dimensions as your final Fireworks document.



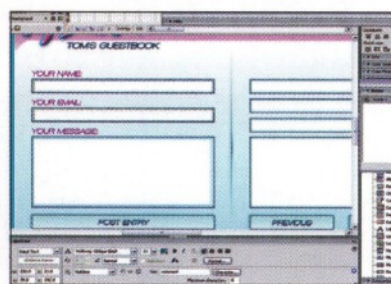
2 Label the first layer code, and leave the frame blank. We'll write our ActionScript on this frame later on. Add a new layer, this time called background. Choose File > Import > Import To Stage, select your mockup PNG file, and check the Import a single flattened bitmap box.



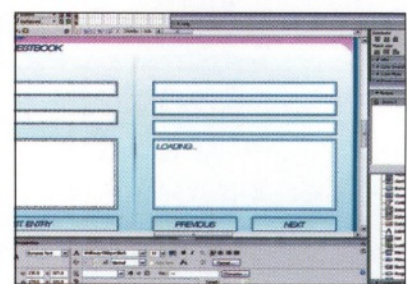
3 With your graphic layout now on the stage, create another new layer above it, with the label 'buttons'. Create a new movie clip called genericbutton, which consists solely of a transparent rectangle.



4 Drag three new instances of the genericbutton clip from the library on to the stage, and resize them to fit over the three button graphics we drew in our interface. Give them the instance names post, prev and next, making sure that they are over the correct button labels.



5 Using the Text tool, draw a text field over the Your name field in the background. In the Properties panel, set it to Input text, and give it the variable value name. Remember to embed the font, and then repeat this step for the other two inputs, using the variables email and comment.



6 On the other side of the panel, draw some more text fields, this time specifying them as dynamic fields rather than input fields. Give them the variable names; aname, aemail, adate and acomment. Don't forget to check the multiline box for both the comment and acomment fields.

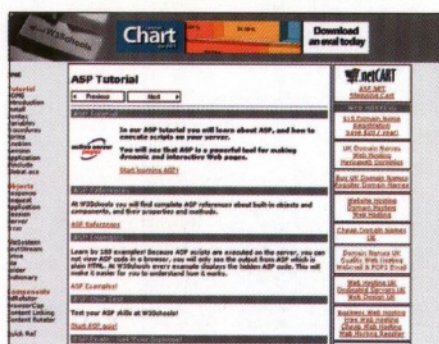
Resources: check out these sites for more on Flash and Fireworks



Internet Tiny Awards

A superb piece of work in its own right, this site features the latest and greatest sites, as well as interviews with top designers.

www.internetinyawards.com



W3 ASP tutorials

The W3 Schools tutorial site carries a huge amount of well-written content, and the ASP section is particularly useful.

www.w3schools.com/asp



Colin Mook

Flash programmer Colin Mook's site is an excellent resource for intermediate to advanced developers looking to enhance their knowledge.

www.mook.org/webdesign/flash

Part 3: Coding the layout of the guestbook

Preparing the guestbook for a movie...

Flash MX introduced a slightly different method of sending and receiving data from a server: the loadVars object. But the old methods – loadVariables and loadVariablesNum – still work. In fact, the are subtle differences in the way these two approaches work and, disappointingly, there are some buggy builds of the Flash 6 player that display some rather odd behaviour when using loadVars. Nevertheless, in the following code we'll demonstrate use of both techniques, by sending and receiving variables from our ASP backend.

On the aesthetic side of things, how many times have the more typographically-obsessed among you wished you had greater control over input fields? With certain fonts, it can look a lot tidier if all text is in capitals. Unfortunately, there is no built-in facility for preventing input fields displaying lowercase letters, so we're going to have to utilize the power of the TextField objects events model.

1 Lines 1–2: First, we use a stop() statement to prevent the movie looping. Next, we define a variable called serverprefix. It tests the application on your local machine, and must contain the localhost path to the guestbook directory. When you're ready to upload, change this value to the empty string.

```
stop();
serverprefix = "http://localhost/
Projects/Net Net/guestbook/";
```

2 Lines 3–5: Here we create a new instance of the LoadVars object. This is Flash's method of handling external data calls and we can give it the name of a function (processData) once we receive our data. Once we've defined the onLoad value, we can load the first guestbook entry via loadvalues.asp.

```
myloader = new LoadVars();
myloader.onLoad =
processData;
myloader.load(serverprefix +
"loadvalues.asp");
```

3 Lines 6–13: The processData function is very simple. The loadvalues.asp page sends us back some variables, which are automatically stashed within our myloader LoadVars object. All we have to do is pick some of them out and store them in root level variables, taking care to capitalize the text values for aesthetic purposes.

```
function processData() {
aname = myloader.aname;
toUpperCase();
aemail = myloader.
aemail.toUpperCase();
adate = myloader.adate;
toUpperCase();
acomment = myloader.
acomment.toUpperCase();
offset = myloader.offset;
totalcount = myloader.
totalcount;
}
```

4 Lines 14–16: This code uses prototype syntax in conjunction with the TextField object's onChange callback. From now on in the movie, whenever any TextField's value is updated, its content is automatically capitalised. Note that if we wanted to do this with just a specific field, we'd remove the TextField.prototype part and replace it with the TextField's instance name.

```
TextField.prototype.onChange
= function() {
this.text = this.text.
toUpperCase();
}
```

5 Lines 17–29: Here we assign a function to the post entry button. We use the other method of retrieving server data, loadVariables, and post our user's data to the signbook.asp page. Then we set up an onEnterFrame loop to wait for confirmation of success, before going back to loadvalues.asp and refreshing the display.

```
post.onRelease = function() {
saved = "no";
loadVariablesNum(serverprefix
+ "signbook.asp", 0, "POST");
this.enabled = 0;
this.onEnterFrame = function() {
if(saved != "no") {
this.enabled = 1;
name = email = comment
= "";
myloader.load(serverprefix
```

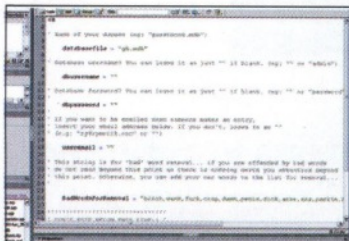
```
+ "loadvalues.asp"); delete
this.onEnterFrame;
}
}
```

6 Lines 30–43: Finally, we set up our previous and next button actions. These two use the offset and totalcount variables provided by the ASP page. This mean we can request the next entry in line. Notice how we set the TextField values to LOADING while we wait to receive our data.

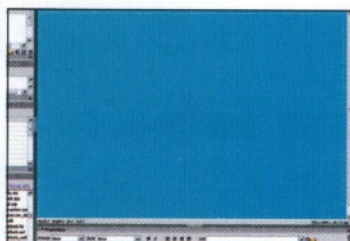
```
next.onRelease = function() {
aname = aemail = adate =
acomment = "LOADING...";
if (offset == totalcount) {
offset = 0;
}
myloader.load(serverprefix +
"loadvalues.asp?offset=" +
offset);
}
prev.onRelease = function() {
aname = aemail = adate =
acomment = "LOADING...";
if (offset == 1) {
offset = int(totalcount) + 2
}
myloader.load(serverprefix +
"loadvalues.asp?offset=" +
(Math.round(offset) - 2));
}
```

Part 4: Applying Dreamweaver

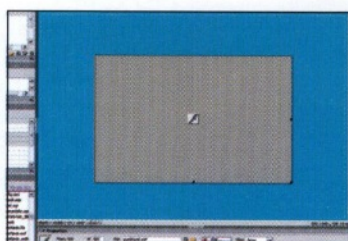
Using Dreamweaver to write the HTML that will display our guestbook movie...



1 Our guestbook uses an Access database called gb.mdb. If you want to be more cryptic, you need to open config.asp. At the top of the page is a variable called databasefile; change the value here to reflect your database's filename.



2 If you haven't opened Dreamweaver, do so and choose New > HTML. We're going to create a basic page to display our Flash movie. Make sure you're in design view and give your page a title in the top field.



3 Hit Page Properties in the Properties panel to alter the page background colour and, using the table tool, create a 100% by 100% table, then choose Insert > Media > Flash to dump our Flash movie into the centre of the page.



4 Save your file as default.asp. If you're running a personal server such as IIS, set up write permissions on the guestbook folder before running the application locally. If not, upload the files and you're ready to go!



Studio MX: Contribute

Learn to build an HTML web site using Contribute as its content management system. **Tom Knowles** leads the way...

For the average small business or personal user, hiring a web programmer to write a custom Content Management System can be expensive. What's more, the selection of freeware CMS packages are often rather complicated to install or, at the very least, assume a certain degree of technical knowledge. This problem of money and complexity can frequently lead the small-

scale designer into favouring flat HTML over any sort of dynamicity and this, in turn, usually means frequent small updates, burgeoning code duplication and, ultimately, client dissatisfaction.

This is where Macromedia's Contribute comes in. Essentially a scaled-down WYSIWYG version of Dreamweaver, Contribute allows you to specify areas of your web site as being editable, and then

provides you with the ability to directly edit and upload any changes you want to make to those areas from within program's user interface — all the while being shielded from the underlying source code.

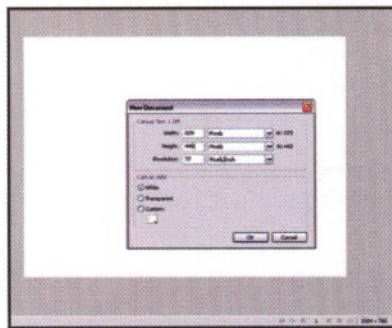
In this tutorial, we are going to show you how to design and build a web site, and we will also demonstrate how to alter the home page without ever having to go anywhere near the raw HTML source code. ■

Part 1: Using Fireworks to design the site layout

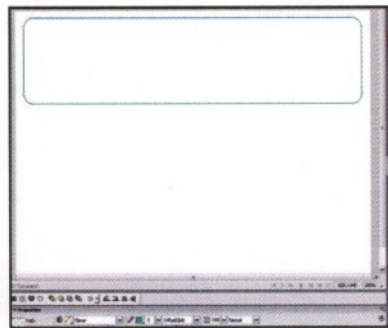
As usual, we're going to design the site layout and graphics in Fireworks...

Too many cooks...

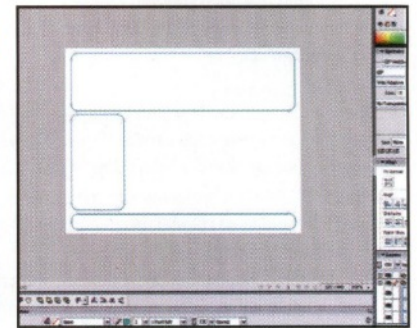
Remember to keep the scope of what's editable in your site as narrow as possible within the specification. The chances are that if a client discovers they can update a part of the site, they will — regardless of whether doing so could cause some problems.



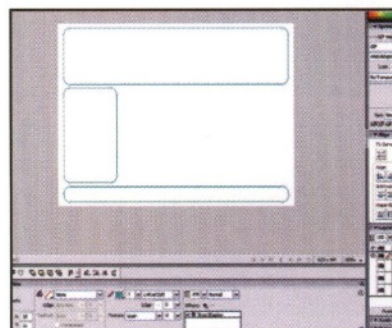
1 We're going to demonstrate just how powerful and straightforward Fireworks is. Open the program and create a new document with a white background and dimensions of 620 by 440 pixels.



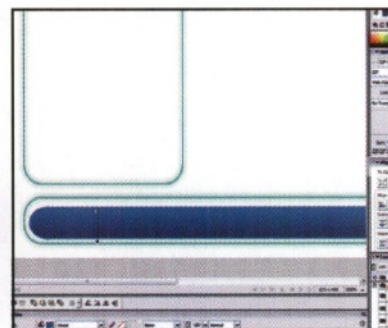
2 Let's start by drawing a rounded rectangle of width 590 pixels and height 137 pixels. You can either use the actual rounded rectangle tool for this, or use the regular rectangle tool and then specify the roundness in the properties panel. Give the shape an empty fill and an outline, as above.



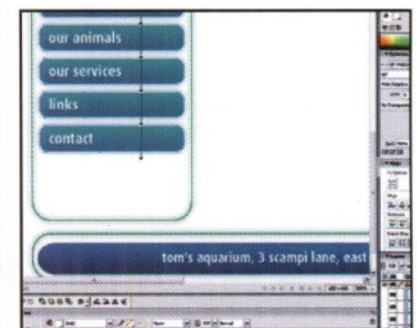
3 We're going to use the rounded rectangle to lay out the site elements, so create two and position them as above. To ensure that the roundness of the corners matches, simply copy and paste your first rectangle, and use the sub-selection tool to adjust the width, without disturbing the curvature.



4 Creating a soft, friendly atmosphere with subtle drop shadows or glows. I personally prefer the look of a 0 pixel drop shadow — it communicates softness without being too in-your-face. To add a drop-shadow, use the effects panel and tweak the various settings to your liking.



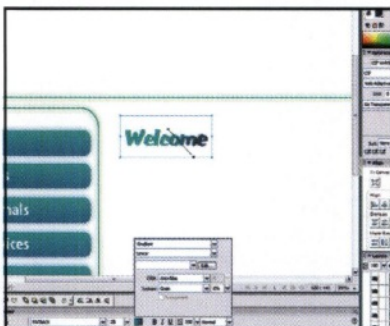
5 With the footer element at the bottom, overlay another slightly smaller rectangle and apply a blue gradient fill in the Properties panel. You can manipulate the gradient direction using the black toggles that appear. To line the two shapes up precisely, make sure to use the align panel.



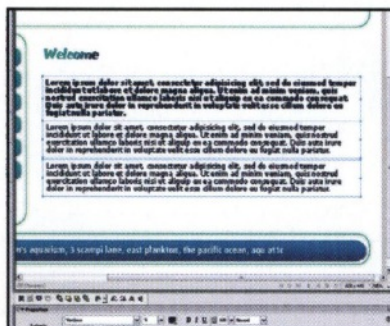
6 As above, overlay some contact details using the text tool. Now, using the same technique of gradient-filled rectangles, create the navigation buttons on the left-hand side of the layout. Again, use a subtle drop-shadow and the align panel's spacing button to keep things neat and tidy.

Part 1: Using Fireworks to design the site layout

continued



7 It's a well-hidden feature but Fireworks allows you to apply gradients to editable text. We're going to use this functionality for our welcome title. Choose a suitable font and then click the fill button in the properties panel. You'll notice there's a Fill options button. Click this to configure a gradient fill.



8 Use the text tool again to write out some dummy content – remember that there's no point using a fancy font for this as most web developers appreciate, you're pretty much limited to Arial, Verdana and Times New Roman, even in this day and age. Now move back to the header area.



9 Draw another rounded rectangle, again with a gradient fill and drop shadow, and place it as above. Since our site is based on water, we've made the gradient a bluish green. Type out some textual content above it – this is usually a good place for email addresses or phone numbers.

Bend and flex

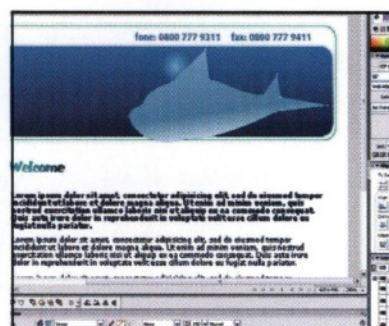
One of the key areas when designing content-managed sites, is flexible, scalable templates. Your site is really as good as useless if the layout breaks as soon as the content changes.



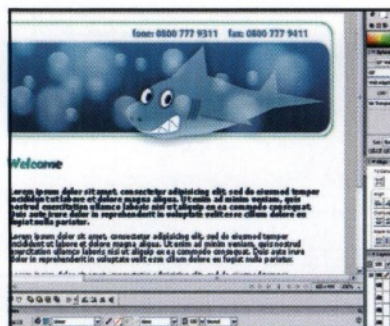
10 A lot of people ask me how to draw the familiar Mac OS X aqua effect – the glass-like sheen that appears on many of Apple's icons. There are a few variations but we'll demonstrate a simple one by drawing some bubbles. Firstly, draw a simple circle, with a 50% alpha setting.



11 Now, overlay another circle but this time apply a vertical gradient fill, going from 100% to 0% alpha. Resize the overlay until its slightly narrower and about half the height of the underlying circle. To complete the effect, duplicate the overlaid circle, move it down and apply a Gaussian blur.



12 To add a bit more fun to the proceedings, let's draw a cartoon shark. As we mentioned earlier, Fireworks makes it fairly easy to make effective illustrations. By using a combination of gradient fills, combined shapes and the sub-selection tool, you can create a decent basic shape in no time.



13 Once you've got your basic shape, you can build up your illustration by adding little embellishments such as gills and nostrils. Remember, you don't have to be an ace illustrator with a degree in fine art; with patience and a focus on detail you can achieve quality results.



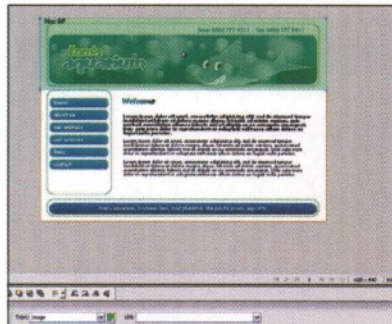
14 Finally, we'll create our header logo. Unfortunately, Fireworks doesn't let you apply an outline to editable text, so choose Text > Convert to Paths followed by Modify > Combine Paths > Union. You can then apply effects and fills as you would with any normal shape.

Part 2: Creating an HTML template

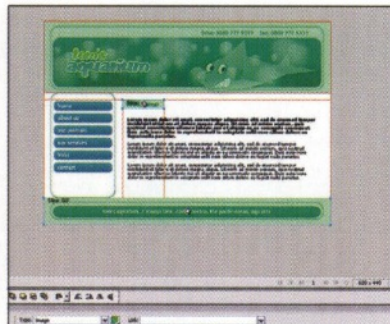
We need to use Fireworks to cut the layout image into usable pieces...

Follow the wizard

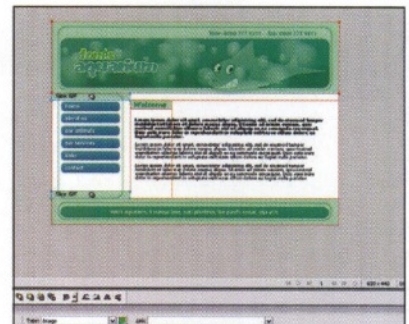
Our site is reasonably simple, so we don't need to bother with any server technologies.



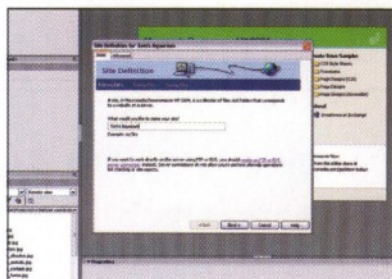
1 Using the Slice tool, draw a block around the header graphic. Choose File > Export Preview and follow the wizard. Remember to check Export selected slices only.



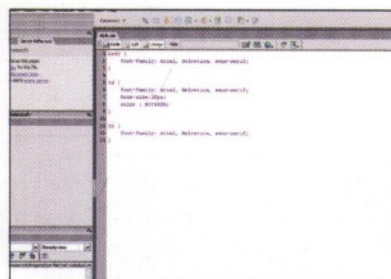
2 You'll now need to go through the individual elements, creating slices and then exporting them. This is all fairly straightforward; the only time we need to be a bit clever is with the navigation section, since we'll need each of the buttons to be separate graphics.



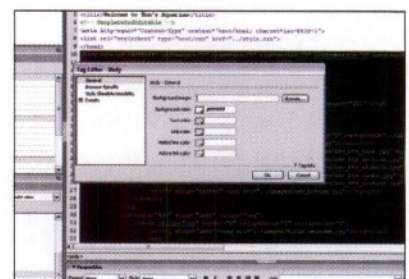
3 To retain the rectangle background, we need to slice up the top and bottom of the navigation and export them as nav_top.jpg and nav_bottom.jpg respectively. Lastly, we need to make a very short slice of the rounded rectangle, which will serve as a table background in our final HTML.



4 OK, now we've got our graphics, let's open Dreamweaver and create a new Dreamweaver site. Once you've set up your site, choose New > Basic Page > HTML Template.

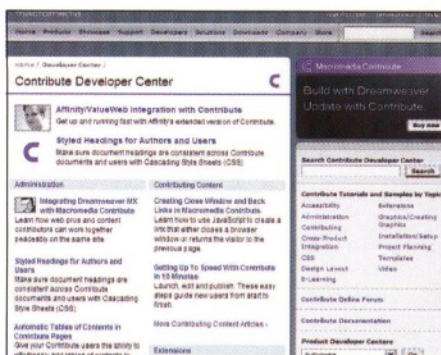


5 We'll cover the main template code later on, but for now save your template as Homepage. Now choose New > CSS Stylesheet and pick the basic Arial or Verdana file. Save it as style.css and then drag it from the assets panel into the head tag of your template.



6 Finally, we need to write the actual HTML to display our site. We'll cover the tabular structure in the next section but, for now, right-click the body tag (in code view) and make sure the background colour is white and the margins are all set to 0.

Resources: Follow the path of MX Masterhood to these sites



Macromedia Contribute Developer Centre
The place to start if you want to learn more about Contribute's wide variety of features.
www.macromedia.com/devnet/contribute



Letterhead Fonts
A great font resource, with some quirky and unique typefaces and some free downloads as well.
www.letterheadfonts.com



stock.xchng
A fantastic site with thousands of free stock photos for use within your Web sites and other multimedia.
www.sxc.hu/index.phtml

Part 3: Coding the tables

There's much discussion about the use of tables versus pure CSS for layouts...

Undoubtedly, CSS is the future and the majority of web sites built in the next five years will probably use it exclusively. Right now, however, it's not fully mature and there's a number of discrepancies in the way that different browsers handle some elements. Add the fact that practically billions of existing web pages currently uses tables, there's no way any future browser will stop supporting them yet.

1 Lines 1–2: The classic 100% wide parent table. Note that as we move towards stricter HTML standards like XHTML, certain old chestnuts such as the 100% high table won't behave as expected. Jakob Nielsen may frown, but if you want your content in the dead centre of the screen, remove your document type definition from the top of your HTML document.

```
<table width="100%" border="0"
cellspacing="0" cellpadding="0">
<tr>
<td width="100%" height="100%"
align="center" valign="middle">
```

2 Lines 3–6: The first row of the nested table contains a cell that spans two columns. This is where we place the header graphic we exported from Fireworks earlier on. Note that if you simply drag the image from Dreamweaver's assets panel, the dimensions are inexplicably left out of the tag. To combat this, right click on the img tag and edit it.

```
<table align="center"
border="0" cellspacing="0"
cellpadding="0">
```

```
<tr>
<td colspan="2"></td>
</tr>
```

3 Lines 8–10: On the next row, the first cell contains another nested table, which is where we'll put the navigation buttons we created earlier. Notice that we've used the background attribute to specify the nav_back.jpg image, which will allow us to add as many buttons as we want without disrupting the layout.

```
<tr>
<td align="center"
valign="top">
<table width="170" border="0"
cellpadding="0" cellspacing="0"
background=" ../images/nav_
back.jpg">
```

4 Now we can just stuff row after row of navigation buttons into the table. We've also sandwiched the button rows between two rows that contain the top and bottom end images. If the relative path to the images seems a bit odd, don't panic. Dreamweaver changes all the paths in a template document to match the location of whatever files you derive from it.

```
<tr><td align="center"></
td></tr>
<tr><td align="center"><a
href="#"></a></
td></tr>
```

```
<tr><td align="center"><a
href="#"></a></td></tr>
<tr><td align="center"><a
href="#"></a></td></tr>
<tr><td align="center"><a
href="#"></a>
</td></tr>
```

```
<tr><td align="center"><a
href="#"></a></td></tr>
<tr><td align="center"></td></tr>
```

5 Lines 18–29: The nitty-gritty: this chunk of code contains the Template delimiting tags which Dreamweaver and Contribute use to determine what you can and can't edit. For now, we've just put some dummy Lorem text in a couple of paragraph tags, but once we've configured Contribute, we can easily change this.

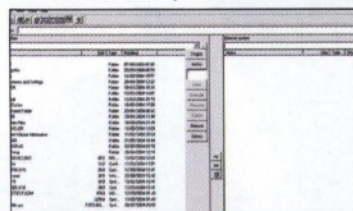
```
</td>
<td width="450" align="left"
valign="top">
<table valign="top"
border="0" cellpadding="5"
cellspacing="5">
<tr><td align="left"></td></tr>
<tr>
<td width="400" valign="top">
<!-- TemplateBeginEditable
name="content" -->
<p align="justify"><b>Lorem
ipsum dolor sit amet,
consectetur adipiscing elit,
sed do eiusmod tempor
ea commodo consequat.</b></p>
<!-- TemplateEndEditable -->
```

6 Lines 30–41: Finally, we place the footer graphic in the last row. That's all that's involved in writing the template. All that is left to do is to save it and then choose File > New from Template, and save the resulting template-derived page as index.htm or another suitable default document name.

```
</td>
</tr>
</table>
</td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table>
</td>
</tr>
</table>
```

Part 4: Contribute

The final step before using Contribute to edit your site is to upload it to a server...



1 If you're running a one such as PWS, IIS or Apache then you can use Contribute to edit the site on your local machine as if it were a remote server.



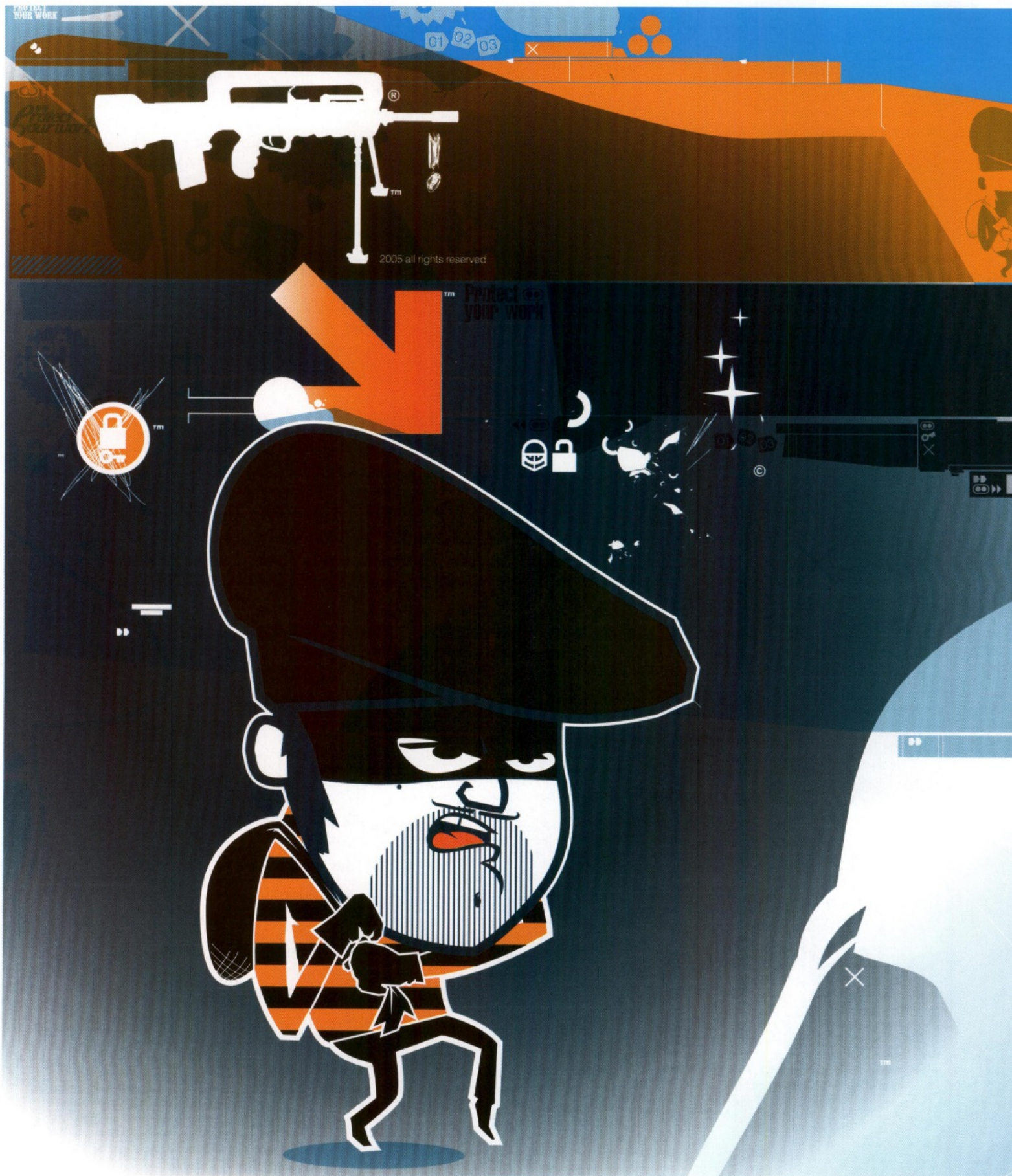
2 Once you've installed Contribute, plugging it into your site is easy. Choose Edit > My Connections and then New. Then follow the wizard. It will ask you for details about where your site is uploaded, your name and your password.



3 Once you complete the wizard, you'll be taken straight to your site. Editing the home page is as simple as clicking Edit at the top. The interface will change to resemble that of Dreamweaver, with the page's editable regions highlighted.



4 All you have to do now is simply edit the content. When you're done, just click the Publish button. Contribute will automatically make the changes and upload them to wherever the site is hosted. That's all there is to it – it's a breeze.



Protect your work

Your work can be copied, its authorship lost along the way, leaving you forgotten and unpaid, so it's well worth putting a few obstacles in the way of thieves. **Karl Hodge** shows you how...

One of the best things about digital images is that they are so easy to reproduce and distribute. But that's not always necessarily a good thing when it comes to your security.

The same goes for web site code, digital video, 3D files or digital art. To get yourself noticed, you need to put your work out there for all to see, and it's much easier to display your work in an online portfolio or a CD presentation. But how do you stop people copying your work and using it for free? Unfortunately, you can't. There is no way to guarantee that your work will be safe from thieves. The only solution is to deter them.

It's most difficult for thieves to work around a blatantly visible watermark – so you could distribute doctored versions of your images with a copyright notice that is clearly displayed, somewhere that's impossible to crop out. Less drastic solutions include using Adobe's new XMP technology in Photoshop CS. This hides a copyright

message within the image file. Retrieve that information and you can easily prove that you are the author of the image.

Between these two poles is digital watermarking. Shipped with Photoshop, Digimarc works by embedding information about you and your work into a watermark within your image and an online database. The watermark can then be read by any Digimarc-enabled imaging program. You'll have to register the plug-in, but with a basic subscription of \$49 a year you can invisibly protect up to 100 images.

The dreaded right-click

Web site code can be made more difficult to read (and therefore steal) by using a compression program such as HTML Shrinker (www.thepluginsite.com/products/htmlshrinker/), but the most secure method of all is to fully encrypt your code or files by using an off-the-shelf script or program. Try Secure Image Free from www.artistcope.com, which will encrypt images so that no-one can link to or

download them from your web site or copy them from a CD.

Your work is particularly vulnerable on the web, where it's easy to right-click on a picture and save it to your hard drive. Luckily, there are a couple of ways to deal with this. Code-based methods using JavaScript and Cascaded Style Sheet (CSS) attributes can disable the way thieves access images displayed online. It's even easier to use a design-based method. With HTML and CSS, you can place an invisible layer over an image, preventing images from the dreaded right-click.

We'll look at these different methods in more depth in the following tutorials. ■



Expertise provided by Luke Whittaker
You can contact Karl Hodge at
www.spodgod.com.com



Files on disc
You'll find the files you need on the cover disc to follow this tutorial.



PROTECT

Protect your work



Part 1: How to hide your content

Do everything you can to deter thieves from stealing your work with this “nested HTML files” trick...

Workarounds

A determined thief could save your entire web site to their hard disk, look in their web cache for images, search your source code to find URLs to files and take screenshots of your images when access to menu tools is disabled. Streaming video can be captured using tools such as VideoLAN (www.videolan.org). Image files with “invisible” watermarks can also be converted to other formats and the watermark may not survive the transition.

```
1
2 <html>
3 <head>
4 <title>Launch Page</title>
5 </head>
6 <body>
7 <a href="#" onClick="window.open('deterrent.html', 'floatingwindow', 'menubar=0,width=800,height=600')">
8 Protected Section</a>
9 </body>
10 </html>
11
```

1 The following easy-to-implement steps create a series of nested HTML files that can be used to help protect your entire web site or just a single section from hacker attack. Start by creating a link that opens a new window without a menu bar. This code can be embedded in a gateway page that leads into your web site. You could also use any navigation link within your site.

```
1
2 <html>
3 <head>
4 <title>Deterrent</title>
5 </head>
6 <frameset rows="14,994" border="0" framespacing="0" frameborder="0">
7 <frame src="invisible.html" name="invisible" scrolling="no">
8 <frame src="contents.html" name="main">
9 </frameset>
10 <noframes>
11 <body>
12 </body>
13 </noframes>
14 </html>
15
```

2 Now create the deterrent.html file mentioned in the code in step 1. This is a frames-based HTML file that you'll use to nest the protected content. The frame itself is invisible to casual users – but anyone who tries to “View Source” will only be able to access the source code of the invisible frame. Insert the code shown here into an empty HTML document before the <body> tag and save it as “deterrent.html”.

```
1
2 <html>
3 <head>
4 <title>Content</title>
5
6 <style type="text/css">
7 @media print{
8 body {display:none;}
9 }
10 </style>
11
12 <body>
13 Protected Content Here
14 </body>
15
16 </html>
```

3 The frameset code references two further HTML files – “invisible.html” and “contents.html”. The first file contains JavaScript and HTML commands that will protect your content and the second file will contain your portfolio, images or whatever you want to protect. Create “invisible.html” by saving a blank HTML file with that name. Next, add the code above to your “contents.html” file between the <head> tags.

IS CD PROTECTION VIABLE?

Fact: As soon as a new method of protecting CD data is announced by the industry, devious hackers waste no time working out how to crack it. Unfortunately, CD protection often works in ways you don't want it to – some methods stop you from playing CDs on systems that have legitimate CD-burning software installed, for instance, while others prevent audio CDs from playing on any computers, full stop. For computer artists, full CD protection mechanisms are less preferable than local file protection.

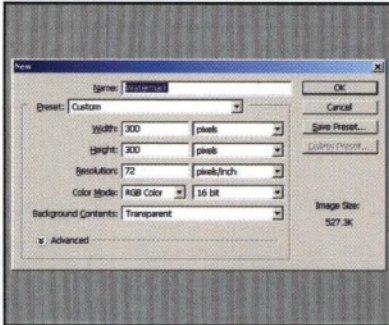
You can password-protect a folder or file quite easily, by putting it into a ZIP or RAR archive, using a tool such as WinZIP (www.winzip.com). Windows users have ZIP archive tools built into the operating system, while Mac users have the ever-popular Stuffit range (www.stuffit.com).

```
1
2 <html>
3 <head>
4 <title>Content</title>
5
6 <style type="text/css">
7 @media print{
8 body {display:none;}
9 }
10 </style>
11
12 <meta http-equiv="imagetoolbar" content="no" />
13 <body oncontextmenu="return false;">
14 Protected Content Here
15 </body>
```

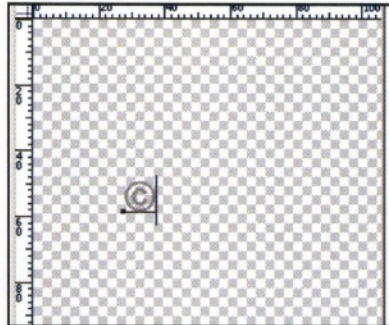
4 The style sheet code shown here prevents users from printing out your page. The next and final pieces of code disable the Image Toolbar and right-clicking in modern browsers – Internet Explorer 6 and Firefox, for example – and should also go in the “contents.html” file where indicated. When combined, these tricks act as a great deterrent, although determined thieves will always find ways around the obstacles you've laid for them.

Part 2: Add a watermark

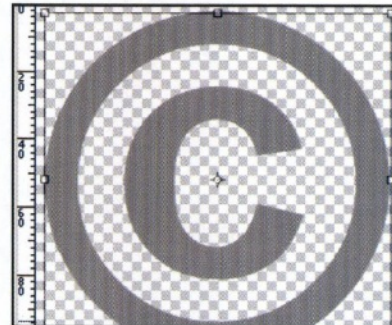
Create a transparent watermark in Photoshop that can be placed over your online images in a Cascading Style Sheet layer...



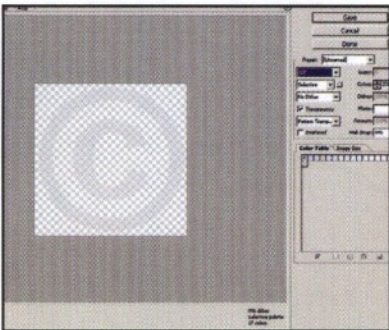
1 Create a new Photoshop document measuring 300x300 pixels with a transparent background. These are arbitrary dimensions, as you'll discover, so feel free to experiment with different sizes, or create watermarks that better fit your own images.



2 Select the Type tool and click within the new document to activate the flashing caret. Hold down the Alt key and type 0169 on the numeric keypad. As soon as you release the Alt key, the © symbol will appear. It's even easier if you're using a Mac. Just hit [Alt] + [G].



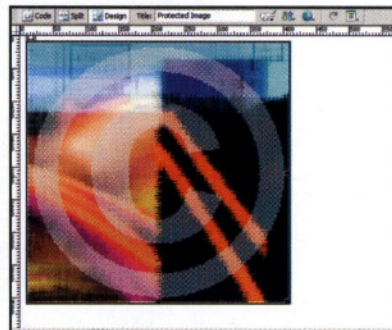
3 Select the symbol and change its colour to a medium grey. Choose a clean Sans font, such as Arial. Select the Move tool to deactivate the Type tool and then hit [Ctrl] + [T] (or [Cmd] + [T] on the Mac) to Free Transform the character. Resize the symbol so that it fills the document.



4 Select the Text layer in the Layers palette and set its Opacity to between 25 and 30 per cent. Under File > Save For Web, select GIF for File Format and tick Transparency. Now select Pattern Transparency Dither under Specify Transparency, and Save file as watermark.gif.



5 Create a new document in Dreamweaver and save it. Click the Layout tab in the Insert menu and choose the Draw Layer icon. Draw a new layer, making it as large as the image you wish to protect. In the Properties panel, set the BG Image path so that it points to the image you want to display.



6 Click within the new layer, choose Insert > Insert Image and browse for the watermark. Save and preview the file in your browser. From now on, any user who tries to right-click and save your image will just get the watermark.

Encryption

Digital data is little more than a collection of zeroes and ones. If you scramble up those numbers in a specific pattern, you'll only be able to view that data if you know how to unscramble it. That's the basis of encryption. Your images or data are rendered unreadable, unless you have a "key" to unlock the encrypted code. You can distribute this key to others, but without it your data will stay unreadable. See www.pgp.net for free file encryption tools.

Combination defence

No one method of protection will stop a determined code or image thief for long. So when it's really important that you deter them, combine as many methods as you can. The web-based methods described in the tutorials here can be easily used together. You can even go one step further and encrypt your pages using a tool such as HTML Protector (www.htmlprotector.com) or search for free encryption scripts at www.scriptsearch.com.

TOP TEN TIPS

1. At the top of an HTML document, before the code section, insert a brief message asserting ownership of the code. Comment this out using the standard HTML comment tags and your browser will ignore it – but it'll still be there to prove your authorship.

2. Hide a second, less visible copyright message in your HTML template. Insert half a page of carriage returns after the closing `</body>` tag and paste it in at the bottom. If someone does steal your code, chances are they'll leave this in.

3. Join an online copyright protection scheme, such as Copyleft (www.gnu.org/copyleft/copyleft.html) or Creative Commons (www.creativecommons.org).

4. Don't include links to high-res versions of your work online. Potential clients only need to get an idea of what your work looks like, so low-res samples will suffice.

5. If you must post your portfolio online, visibly watermark every image. It sounds drastic, but this will definitely stop others from printing or publishing your work in media you have no control over.

6. If you feel the watermarking method described here will spoil your images, use a blank image file instead. When users right-click to save, they'll download the blank image instead of your picture.

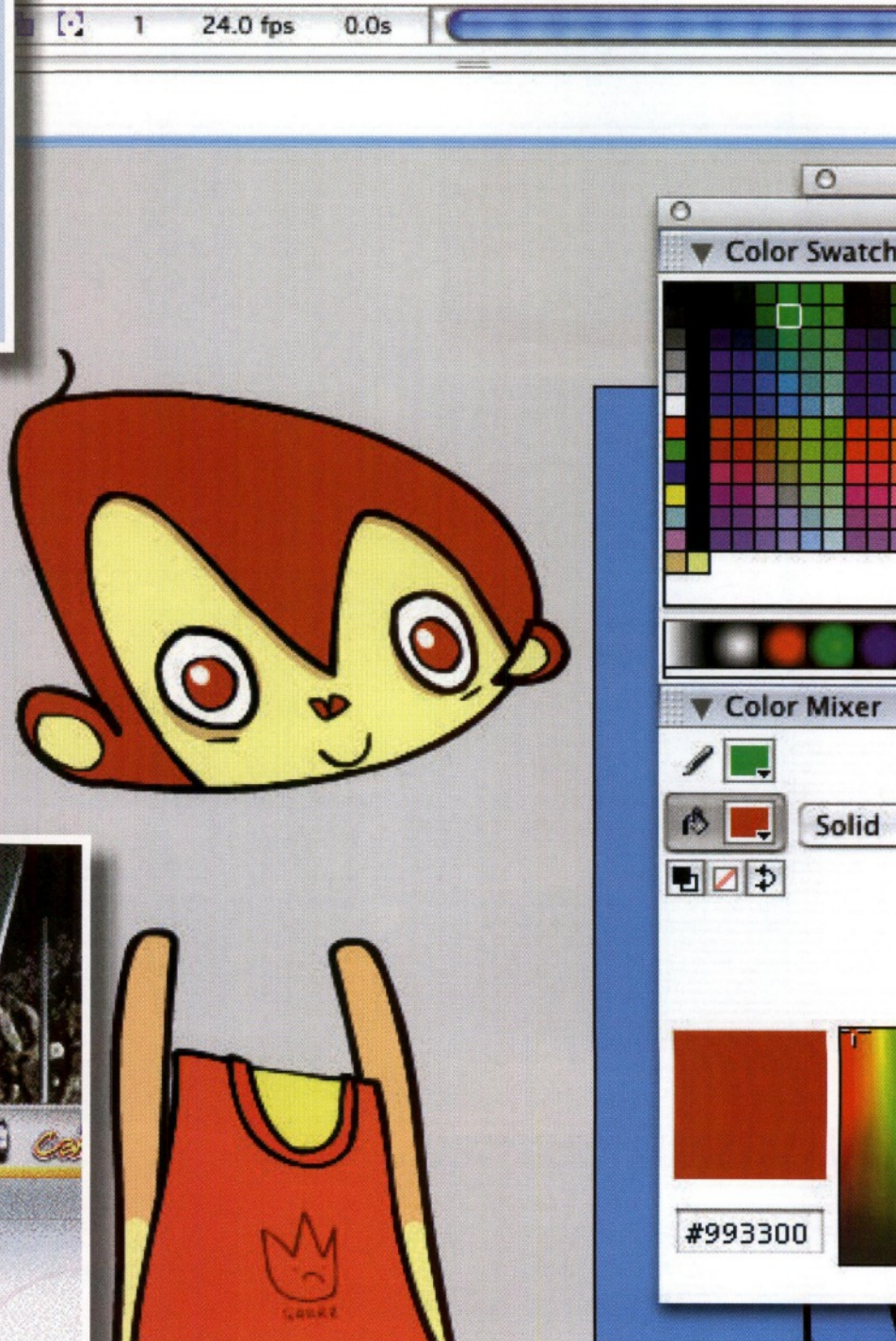
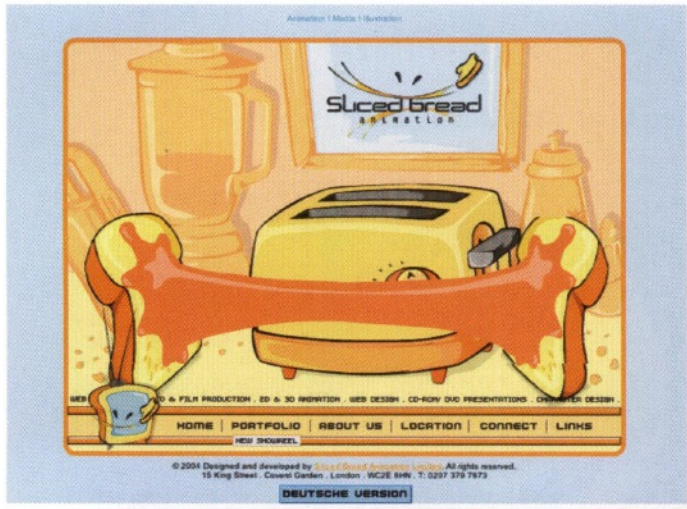
7. Digital Rights Management (DRM) is built into modern streaming media servers and producers to protect your content. Find out more at www.info-mech.com/what_is_drm.html.

8. You can insert the © symbol into your HTML documents using the ASCII code "©". Ensure it appears visibly alongside a short

message on every page containing content you wish to protect.

9. Rename image files so that they're difficult to recognise in a file list. You can batch rename files in Photoshop or Fireworks progressively, replacing intuitive names with serial numbers.

10. When distributing your portfolio on CD, use a multimedia authoring tool such as Flash MX 2004 or Director MX to create a standalone presentation. Content is harder to extract from this kind of file.



3

Flash MX 2004 techniques

There's no better way to gain new skills and inspiration than through practical experience – these tutorials show you that Flash finesse is just a few clicks away...

- | | |
|-----|---------------------------|
| 112 | SHOWCASE: Flash |
| 116 | PROFILE: Electrotank |
| 118 | Flash techniques |
| 120 | Building Flash components |
| 126 | Information management |
| 130 | Storyboarding a movie |
| 134 | Rotoscoping video |
| 139 | Make an icon font |
| 143 | Type in motion |
| 148 | Typographic animation |
| 152 | Better characters |
| 156 | Character animation tips |
| 160 | Lip and facial synching |
| 164 | Countdown Clock |
| 168 | Sliding Puzzle |

SHOWCASE: FLASH

See more of the best Flash content at www.macromedia.com/showcase/flash



Company > Summit Projects Project > Nike Golf

Nike Golf gave Summit Projects just 30 days to completely rebuild nikegolf.com. Summit turned to Flash MX as the most powerful solution for developing rich online content to meet all of Nike Golf's goals and get the job done on time. Working in conjunction with Nike Golf Image Design, Summit created a compelling web presence.

Jason Herkert, Senior Designer at Nike Golf says: "Summit used Flash MX as a technology capable of achieving our bottom-line objectives: a cool, uncluttered look and feel that would attract a younger audience and appeal to our older, core customers, rich visuals and compelling product stories, a refined navigation and corroborative marketing capabilities. The end result is a site that effectively promotes our products and our love of the game."

[w] www.nikegolf.com www.summitprojects.com

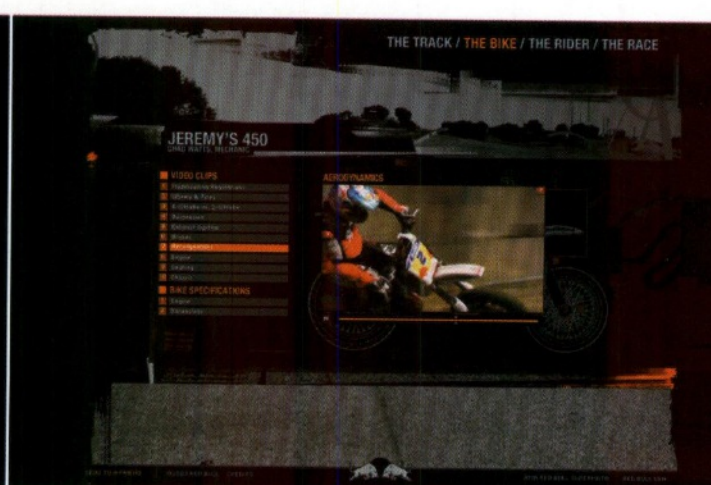


Company > Odopod Project > Red Bull Copilot: Supermoto

Red Bull commissioned US-based agency Odopod to transform Red Bull's video content into an interactive form. David Bliss, a co-founder of Odopod, and Technical Director of the Red Bull Copilot: Supermoto site, talks about interactive video on the web and why Macromedia video was the only viable solution for this project:

"We never really considered using any technology other than Macromedia video, mainly because we could readily wrap logic around the video in Flash. Because we can put the video into Flash, we had great programming control, could develop our own interface elements, controls and cross-links. We could seamlessly integrate all the Flash content on the site with the video. There are four sections to the site and each one is 85-90 per cent video."

[w] www.odopod.com www.redbullcopilot.com



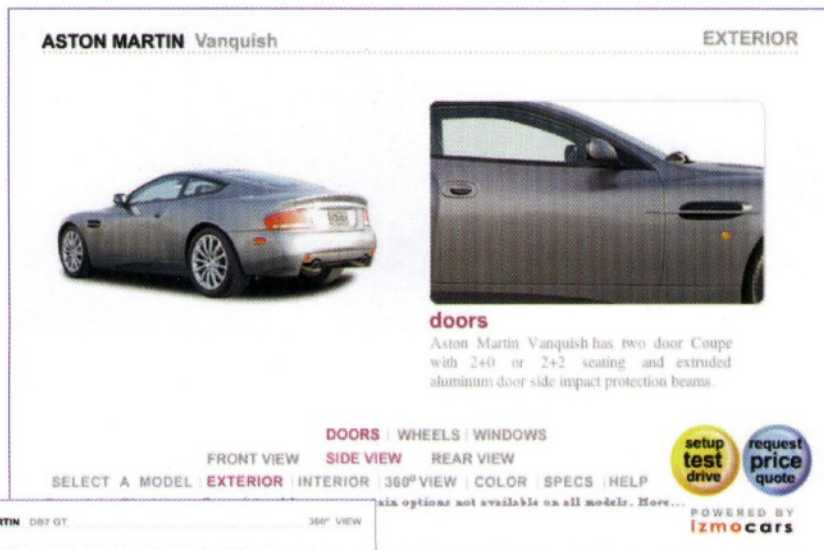
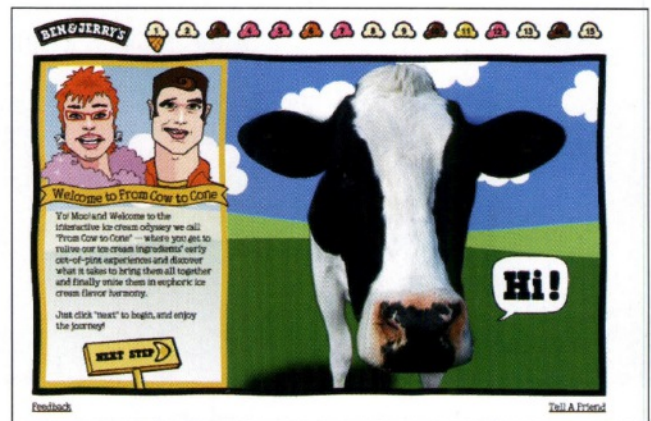


Company > Ben and Jerry's Project > From Cow to Cone

From Cow to Cone is an online application at Ben and Jerry's web site, providing the reader with a virtual tour of the ice cream giant's factory. The tour uses Flash to blend audio, video, and animation into a seamless user experience. Flash video is mixed and synchronised with text and animation. Users can navigate interactively to get behind the scenes at Ben & Jerry's for interesting insights into a company that makes headlines for the right reasons.

Bob Kilpatrick, Senior Web Developer at Ben & Jerry's, talks about embracing Flash video: "Without Flash, this project would have been out of our reach. We wanted users to sit back and be virtually transported to Vermont for an entertaining and informative experience without clicking around — not an easy thing to accomplish on the web. Before this project, we were using Flash to simply animate cows. Using our existing Flash skills, we were able to ratchet up our site with Flash video to create a far richer experience than we've ever had — without spending any dollars to alter our infrastructure. As an in-house developer, I'm not able to be an expert in everything, so Flash and the advent of Flash video has been a very good way for me to grow professionally. Flash makes me feel like I can build anything."

[w] www.benjerry.com



Company > Izmocars Project > izmocars.com

Designing life-like animations with Flash, Izmocars has created virtual tours of virtually every car going. "Our goal is to capture the emotional appeal of the automobile and provide a rich online experience to prospective customers," says Tej Soni, President and CEO of Izmocars.

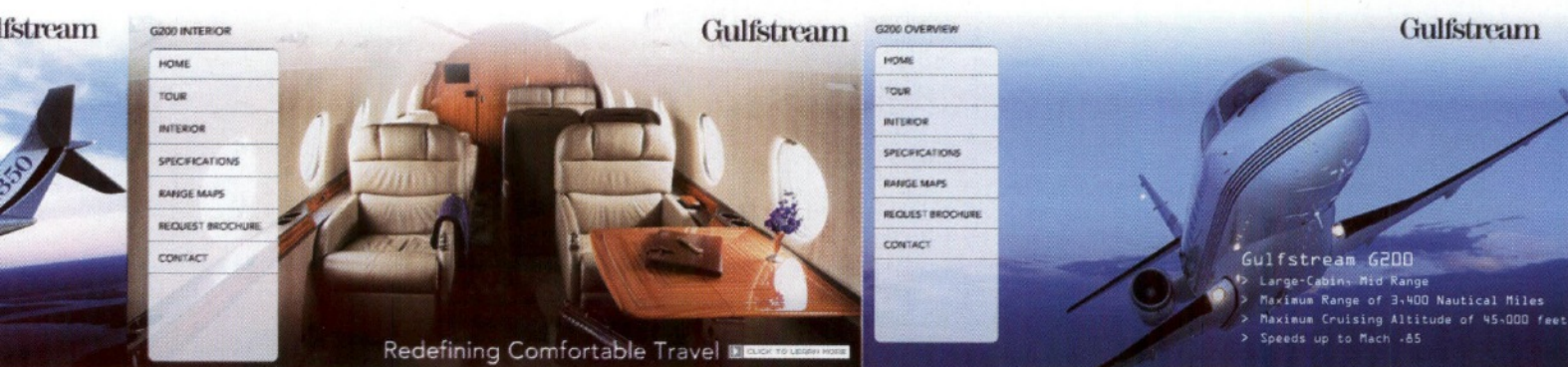
Each year, Izmocars creates animated tours of more than 400 cars generating more than 8000 Flash files. The animations, based on actual photographs, display realistic exterior, interior and 360-degree views of vehicles and also feature a dynamic colour selector option which changes the colour of the vehicle in real time with the click of the mouse.

According to Soni, by using Flash, Izmocars can develop animated car tours faster and at a lower cost than using any other technology. Plus, the experience is more seamless and visually engaging. "With Flash, we can develop these projects in 30 per cent less time than it would take to develop content using any other product. And thanks to vector technology and the Flash Player, the experience loads quickly and seamlessly."

[w] www.izmocars.com

SHOWCASE: DIRECTOR

See more of the best Director content at www.macromedia.com/showcase/director



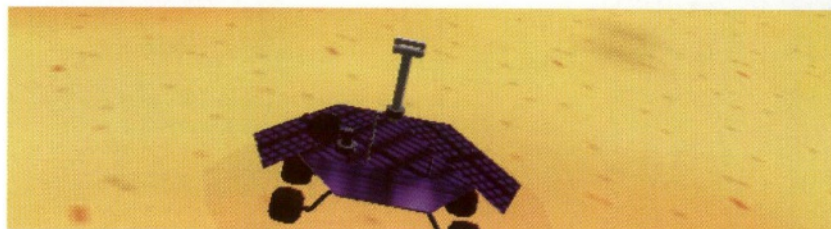
Company > Gulfstream Aerospace Project > Gulfstream Aerospace DVD

Private Jet company Gulfstream Aerospace has manufactured more than 1400 aircraft for corporate, government, private and military customers around the world. When its sales force needed a new, engaging and easy-to-navigate marketing and sales tool, they wanted more than just a leave-behind pamphlet. What was needed was a combination of video and other interactive assets, something that raised the bar of sophistication when the sales executive was meeting with a client.

"The purpose of it is really to provide a direct marketing tool and a sales tool, so that the sales force can sit with a client in front of a high-quality video and navigate through these different parts of the video in an immersive presentation," says Will Dent, Manager of interactive marketing for Gulfstream Aerospace.

He continues: "This version of Director completes the MX 2004 product line, as far as development and integration. Because I know we're going to do more DVD/CD-ROM web site projects, what this means is truly being able to lever and repurpose all my materials including code, components and .swc files, which greatly improves our workflow."

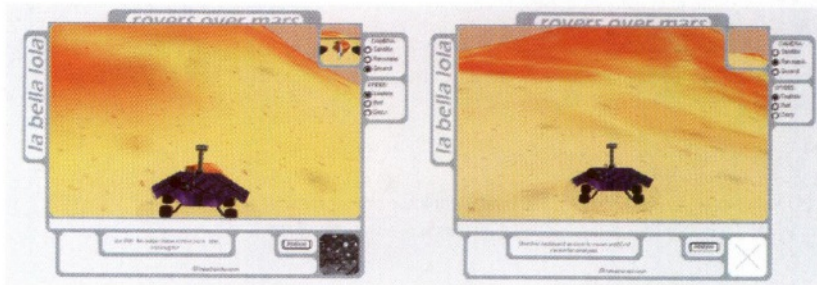
[w] www.gulfstream.com



Company > La Bella Lola Project > Rover Over Mars

La Bella Lola uses Macromedia Director MX to deliver an interactive game that enables users to drive a NASA Rover around Mars searching for traces of water. The result is a realistic Shockwave 3D game.

[w] www.labellalola.com/mars/



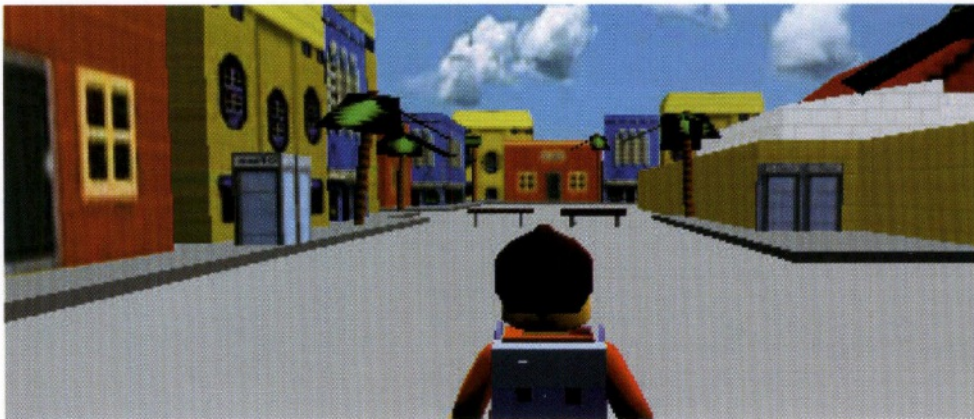
Company > Giant Wheel Production Project > The Lord Of The Rings: Build Your Own Orc

The official *The Lord of the Rings* web site contains unique companion modules that feature exclusive behind-the-scenes content. The 'Inside the Effects' module features an interactive 3D Build Your Own Orc application that simulates the results produced by the Orc Builder application used for the actual movie production. The application also simulates the motions built for the characters by the special effects experts.

The application was built in Director by Giant Wheel Productions and uses Shockwave 3D extensively.

[w] www.lordoftherings.net www.giantwheelproductions.com





Company > Templar Studios LLC Project > LEGO Studios Backlot Game

The LEGO Company, based in Billund, Denmark, is a world leader in providing quality products and experiences that stimulate children's creativity, imagination and learning. The LEGO Studios Backlot game introduces kids and parents to the LEGO Studio product line through an interactive gaming experience. Players explore a simulated movie studio where they can get a job, learn about filmmaking, and find clues to help solve a mystery.

The game was built by Templar Studios LLC using Director and integrating Shockwave 3D technology. The brief was to create a high-quality 3D game that has console quality game-play that could be delivered via the web.

[w] www.lego.com/studios/backlot/default.asp www.templar.com



Company > Skyworks Technology Project > Candystand.com

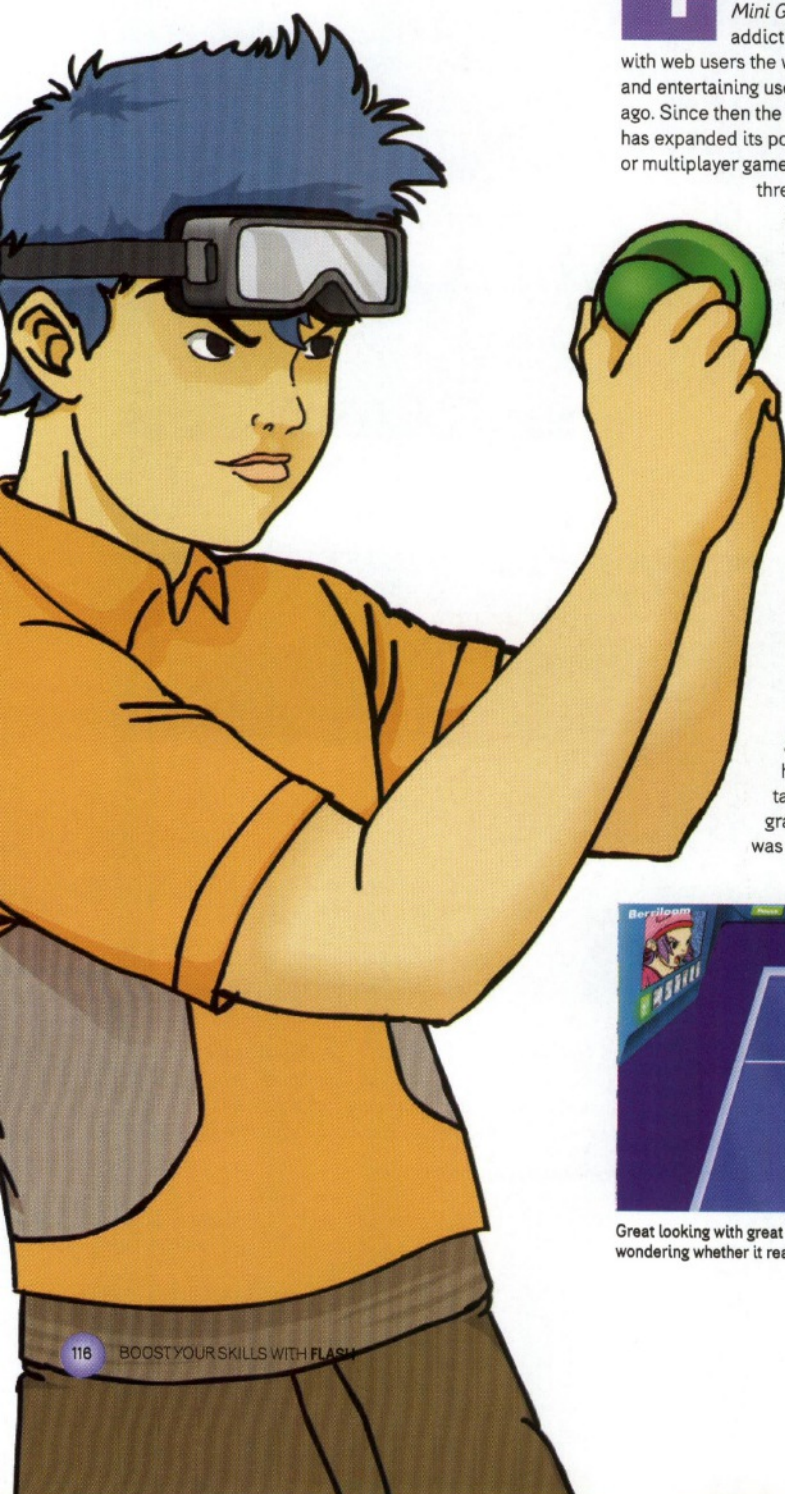
This Kraft Foods site attracts more than 20 million visitors each month. Why? Because of the outstanding Director-based games. You'll find everything from American football to mini golf, all with a sweet or biscuit-based theme. Some truly addictive games here.

[w] www.skyworks.com www.candystand.com



Electrotank

"Our number one goal is to create great games." Electrotank's Robert Firebaugh shares the secrets of successful Flash games...



For a studio's first production to experience the success of Electrotank's *Mini Golf* is no easy task. The simple but addictive online game was an instant hit with web users the world over for its great gameplay and entertaining use of Flash. But that was three years ago. Since then the small game development outfit has expanded its portfolio to over 15 strategy, puzzle or multiplayer games, won several awards and written three books on Flash game development.

Think-tank

Electrotank was founded by three game addicts — Jobe Makar, President; Mike Grundvig, Chief Technical Officer; and Robert Firebaugh, Creative Director. Based in North Carolina, the outfit has a small group of designers, and drafts in freelancers for sound production and bigger projects.

For Robert Firebaugh, Electrotank has been a great way to fuse a successful career in new media with a love of games. Founding the studio with Jobe and Mike meant combining his own illustration and animation talents with Flash's powerful graphics tools. "When we started, it was just another hobby, a way to



Not just a free game on a PC, *Solitaire* has been taken to new heights with a collection of card-based games in Electrotank's Regal Solitaire series. "Surprisingly, there is a lot of strategy that goes into playing *Solitaire*," says Robert

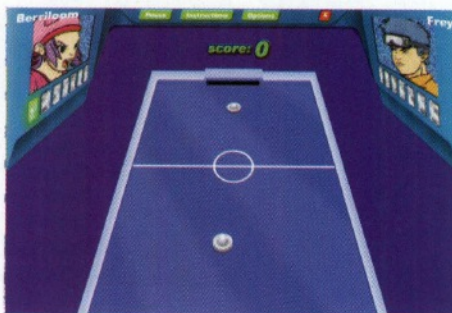
channel my creative juices because my day job at the time was not very challenging. I didn't have any aims other than having fun and creating great games."

Over three years later, the trio's aims are still the same, with the added challenge of finding financial stability in an unpredictable industry. It's risen above the problems faced by other studios by offering game design or existing licences to clients as well as giving leisure gamers the chance to upgrade from free demos of their games to full versions for a nominal fee.

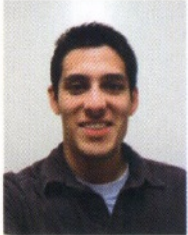
Multiplayer gaming

A recent addition to the business plan is a multiplayer server product developed by Mike Grundvig. ElectroServer is a piece of software that links an unlimited number of clients together. All of Electrotank's own games are based on this server technology and now they are offering it to clients. To date, the proprietary software has attracted clients such as Nickelodeon, MiniClip and Rubik. "ElectroServer makes chatting and multiplayer games possible by intelligently exchanging data between clients at super speeds," Robert explains. "The majority of socket-servers out there have a problem with stability at high numbers of users. ElectroServer has been tested to over 5,000 simultaneous users with no stability issues." A 20-user version is currently available as a free download from Electrotank's site.

But it's not just the backend that makes Electrotank one of the hottest names in game development. The range of games, styles and sounds in its portfolio continues to attract gamers to its web site. "We don't like to stick with one style — the type of game dictates



Great looking with great sound effects, *Air Hockey* will leave you wondering whether it really was created just in Flash — but it was



electrotank

Co-founder and Creative Director, Robert Firebaugh

how each game will look, sound and feel," says Robert. "The projects we enjoy working on most are those with a distinct artistic style or vibe, because you have better odds of creating a game that no one has ever seen, making it more appealing."

So what does make a good game? "That's like asking the secret of the universe!" he exclaims. "A good game is one a player will enjoy. The only way you will know if a player will enjoy it is to put yourself in their shoes. Understand who is going to play your games and design it around pleasing them. People play games to have fun so don't set them up to lose."

Thorough planning

Electrotank takes its planning for each game seriously, with no development work until the idea and scope of the game is thoroughly drafted out. Then time, budget or technical issues are taken into account before building a test engine to see how it will work. It's only when this has been approved that the art and animation work begins, followed by the sound effects. Finally, it gets tested thoroughly. "Get feedback from as many people as possible," recommends Robert. "After working on a project for a while, little things can slip by that someone with a fresh look will spot."

All its games are designed in Flash and use ActionScript to connect to the server. It's not just the latest version either, the art team uses a combination of Flash MX 2004 and Flash 4 due to some features in older versions that help with art production. But for development, it's MX all the way. "With newer versions of Flash, the possibilities are much greater than when we first started," says Robert. "The online gaming industry has grown, there is definitely a lot more competition now."

But as far as he's concerned, the industry is still in its beginnings. "The way the user interacts with a mouse and keyboard through a browser is still relatively new," says Robert. "I feel we still have a way to go before massive improvement is made in online game design." ■



A sneak peak at a brand new project from Electrotank towers. *Tankcore* is a real-time multiplayer warfare game. In Robert's words, "It'll be a real jaw-dropper!"



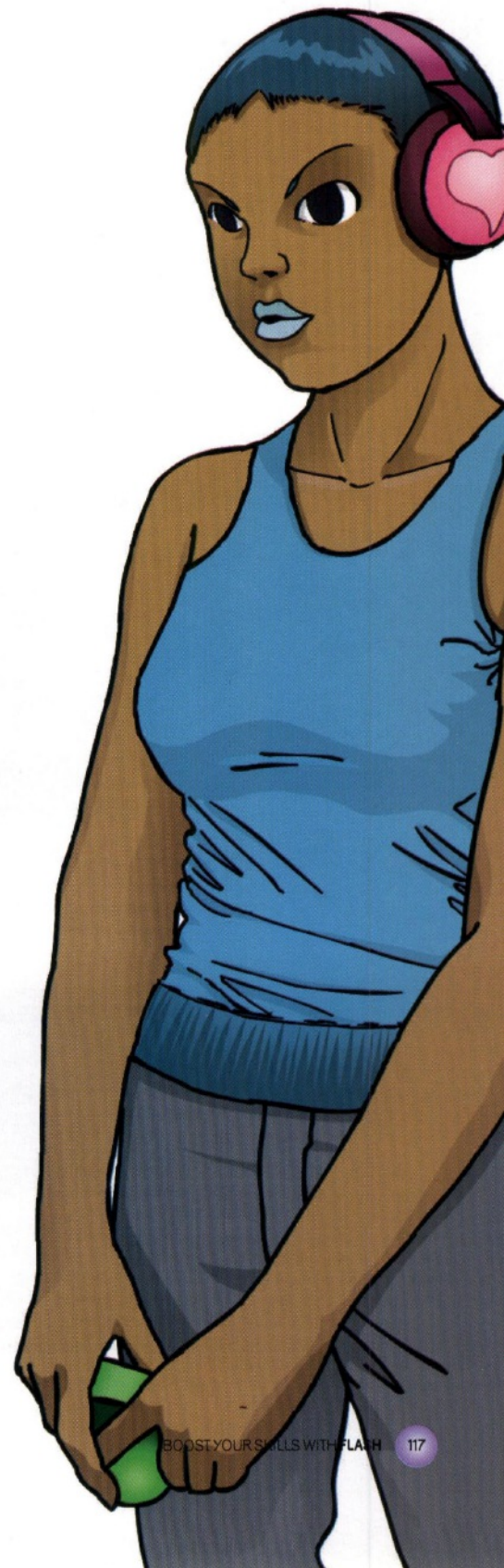
Mini Golf was Electrotank's first online game production and was a massive hit. See how the simple vector shapes compare with the more advanced characters of *Air Hockey*



Fruit Smash is a take on the traditional stacking games. Move fruits into lines to make them disappear and beat the clock. With great use of Flash, it's colourful and highly playable



You can contact Electrotank via its web site at www.electrotank.com or call 001 919 609 0408. For more information on its ElectroServer software and to download the free version, see www.electro-server.com





Graphics: Flash techniques



The components to complete this tutorial are on your cover disc.

Make your site move with some well-placed, easily executed Flash animation

Flash animation is often associated with objects moving at great speed. They normally fly onto the screen from all directions, coming to a halt with a little bounce as if they have hit a brick wall. In the right place it's an effective technique. However, not every project requires this level of movement. Sometimes subtlety is called for.

If you have been following the past two tutorials you'll probably already be aware of

the test site we've created at www.brick-on-the-egg.co.uk/net/webgraphics. It's an HTML site with very little opportunity for radical Flash animation.

However, we were keen to give the site some movement and decided that the best opportunity lay with animating the large image of clouds at the top.

As this is quite an extensive area of the web site, it would be crazy to employ an animation that would be either processor heavy or very distracting. Bearing this in

mind we decided to create a gentle looped scroll, where the clouds continually scroll to the left without any obvious joins.

To achieve this effect you need to create an image that can tile or, in other words, the left side of the image needs to join seamlessly with the right hand side. This can be tricky and that's reason we've devoted most of the tutorial to it. Once you have the image, it is quite a lot simpler to set up a Flash movie that will scroll the image in a permanent loop. Have fun! ■

Part 1: Create a tileable image in Photoshop

Abstract images are easier to tile seamlessly...

RESOURCES FOR FLASH SITES

Flashkit

Not the source of inspiration it used to be but still a very useful resource for everything and anything to do with Flash.

www.flashkit.com



Royalty free

Thousands of royalty free images for just a few pence each. Even better, upload your own photos and get free download credits.

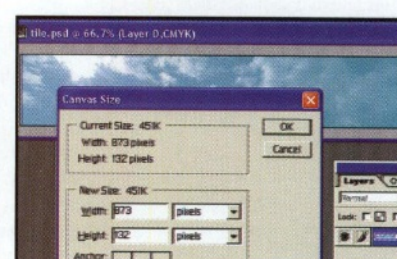
www.istockphoto.com



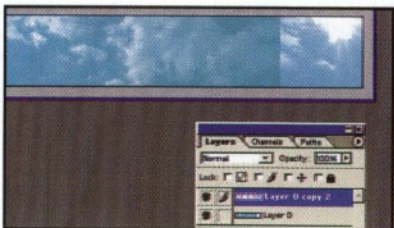
1 Firstly, choose the image you want to scroll. We've used a picture of some clouds from istockphoto.com (see resource 2). You can, of course, use any image, but the more abstract the image the easier it is to tile seamlessly.



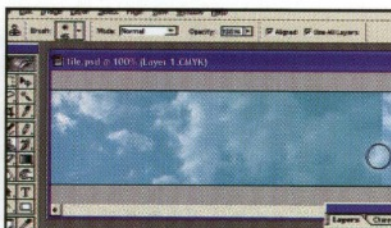
2 Now crop the image to the correct size. Select the Crop tool and enter the width and height required into the options palette, 718x132 pixels in this case, set the resolution to 72dpi. Click and drag on the image, adjust, and when happy hit the Return key.



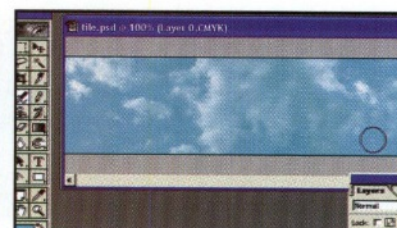
3 Save a selection the same size as the image by choosing, Select > All, then Select > Save Selection. Name the selection and make sure New Channel is highlighted. Then choose Image > Canvas size and add 150 pixels to the right of the canvas.



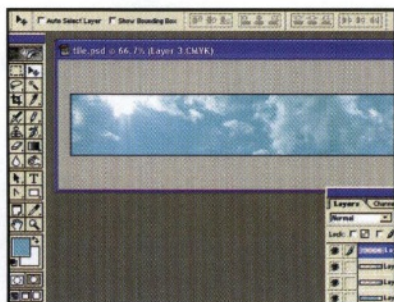
4 Right-click on the image layer and create a duplicate. Select the Move tool, hold down the Shift key (to constrain), and drag the new layer to the right, until its left edge aligns to the right of the image below. This is the join we will now make appear seamless.



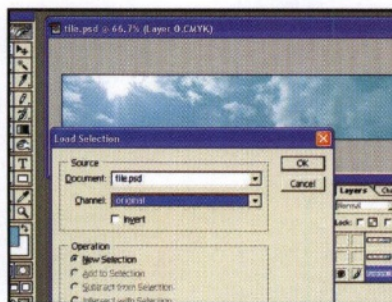
5 Create a new layer, select the Rubber stamp, and make sure the All layers option is selected. Choose a medium brush, hold down the Alt key and click on one of the clouds, release the Alt key and then start airbrushing cloud detail over the join.



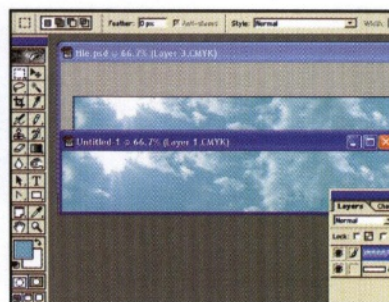
6 You can also use the Airbrush tool. Select a brush, adjust the opacity, and select a colour by pressing Alt and clicking on part of the image, now airbrush gently as required over the join. Repeat steps five and six until the join is totally disguised.



7 Move a copy of the join to the left of the image. Go to Select > Load selection and load the saved selection, press Ctrl + I to invert it, then select Edit > Copy Merged and Edit > Paste. Use the Move tool to drag the new layer to the left.



8 That's it, the part of the image defined by the selection you saved earlier, should be fully tileable. So load the selection again (see step 7), select Edit > Copy Merged, then go to File > New, name the document, hit OK and then select Edit > Paste.



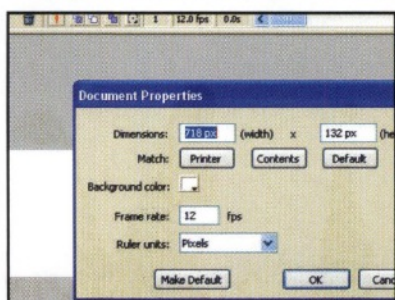
9 Finally, select File > Save for web. Choose the PNG format and hit OK, now name the file and choose the correct location, before clicking Save. Follow the next tutorial to find out how to bring this image into Flash and create the scroll animation.

Make an impact

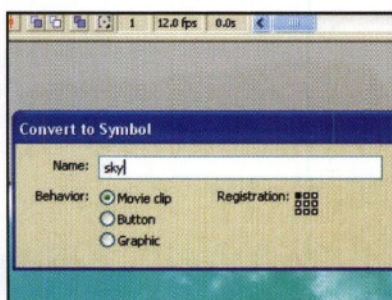
You don't need to use a lot of different colours to create an impact with your site. In fact, the fewer the better. Start off with just one or two contrasting colours, only adding additional, carefully selected ones if it become essential.

Part 2: Create a looped scrolling animation

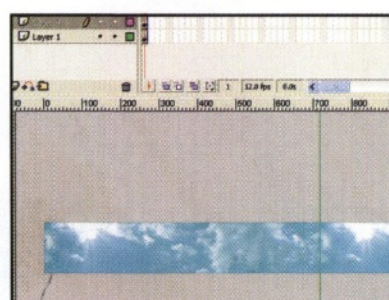
Using a tileable image in Flash...



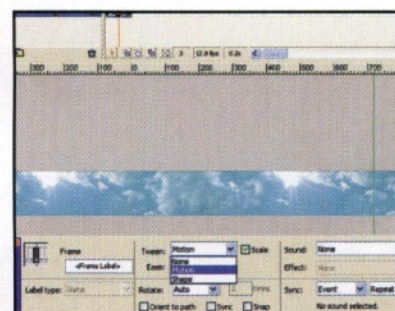
10 Open Flash and Select File > New, double-click the framerate (12fps) at the bottom of the timeline palette. A properties box should open, change the width and height to that of the tileable image you created previously and increase the framerate to 30.



11 Go to File > Import > To stage and select the PNG file you saved in the previous steps. With the image still selected on the stage convert it to a symbol by pressing F8. Name the symbol, choose the graphic option and hit OK.



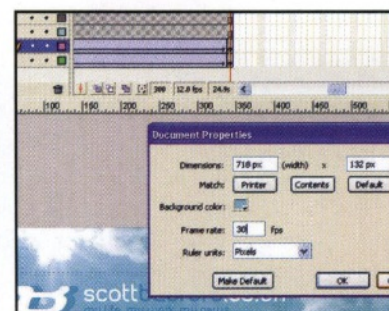
12 Make a new layer, click the symbol you just created and then go to Edit > Copy. Click the first frame of the new layer and then select Edit > Paste in place. Move the symbol to the right, until it aligns perfectly with the right of the image below.



13 Click the top of the timeline to deselect the layers. Press F5 to add extra frames. Click the last frame of each layer and press F6 to insert keyframes. Click on the first key frame of each layer and select Motion tween from the properties palette.



14 Select, in turn, each of the last key frames and move the symbol on the stage to the left by a distance equal to their width (718 pixels in this case). Make additional layers and create or import any other elements such as logos and copy.



15 Click the top of the timeline (between the first and last frames) to deselect the layers. Hold down F5 and add around 300 extra frames. Press Ctrl + Return to both preview and publish the SWF file. Add or remove frames to adjust the speed of the animation.



Illustration by Lasse Skarbovik. Contact via The Organisation [e] organise@easynet.co.uk



Building Flash components

Create fully customisable interface elements from scratch...

Over the following pages we show you how to build two components, combine them into a colour palette and address the colour properties of a MovieClip. The components will have all the standard features of professional components, including inspectable properties. The designs are based on the standard component template used by professional developers all over the world. Professional component design is based around the new class model introduced in

MX 2004. This new style of development means that you can build beautifully crafted components that any designer would be proud to present.

The example we're building will be a bridge between design as traditionally done in Flash and the new class-based design method. One of the criticisms of the new model has been the file size, but we will build our component with file sizes that are similar, if not smaller, than old-style MovieClips. Let's start building. ■



Expertise provided by David Aaron. You can contact David at david.aaron@thatplanet.net



The files needed to complete this tutorial can be found on the cover disc.



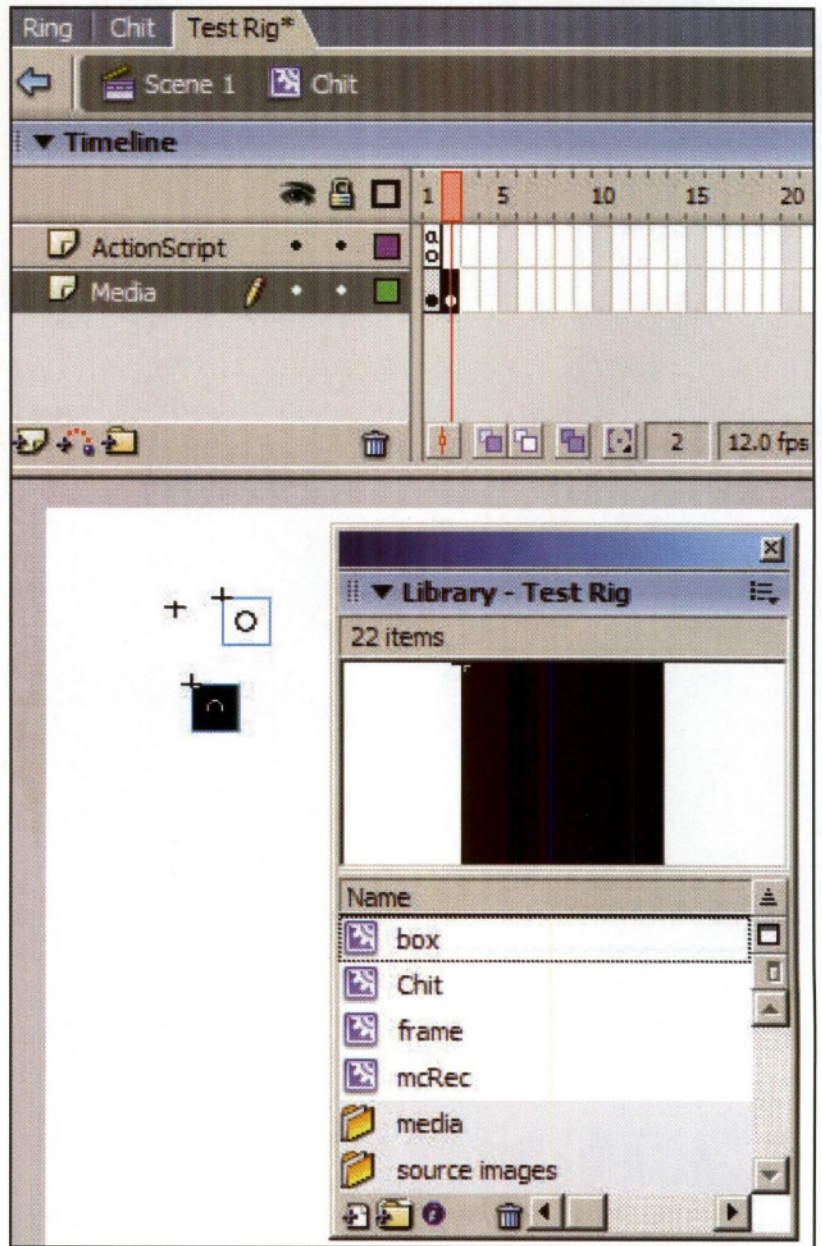
Part 1: The theory

Some important points before you get started...

Most components have a few basic requirements:

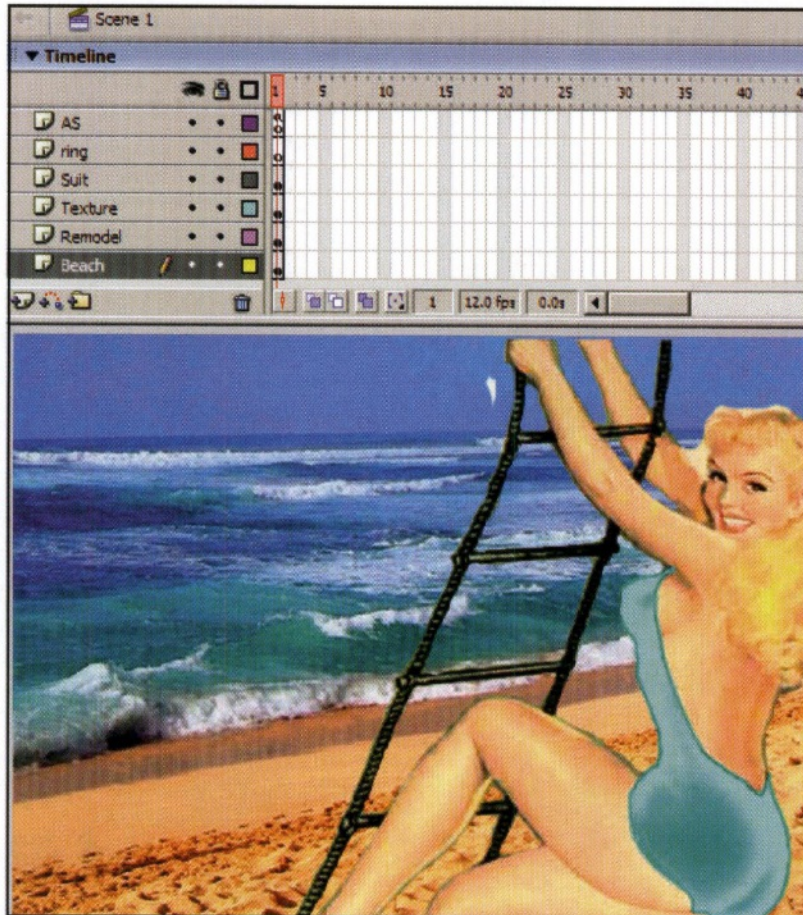
- A MovieClip with two layers and two frames.
- The first keyframe in the first layer is labelled ActionScript, and contains nothing more than a 'stop();' script.
- The first keyframe in the second layer has a bounding rectangle to help Flash draw the component in the library. You don't need to do this if you're not attaching media with code. The second keyframe (see image) in the second layer has all of the media you'll be dynamically attaching.
- The MovieClip name should start with a capital letter. The linkage and class names should be set to match the MovieClip. Class names do not end in '.as'.
- MovieClips do not become components until you apply the component definition (right-click on the bottom of the list).
- Components are not complete until you have either compiled them or converted them into 'SWC' format. We will be compiling components in this tutorial.
- All components have a standard template. You will not need to use all of the template for all of your components.

Note that there are lots of other tutorials that cover the template in more detail on the Macromedia web site. This is simply what you need to know to get started.



Part 2: Laying out the stage and component media

Begin by creating your symbols and getting everything ready for the code...

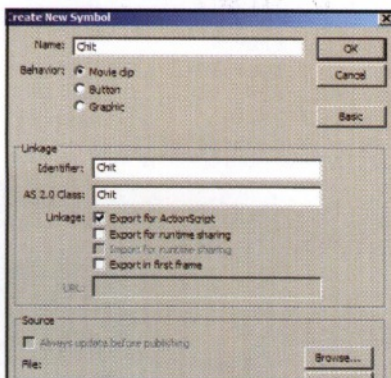


1 Create a movie and save it in a directory by itself. Create six layers. Get the beach and remodel from the completed example's library on the cover disc and drag them onto the layer with corresponding names. You need to build the ring component so that layer remains empty. Drag the texture map MovieClip onto its layer and position it over the remodel's bathing suit. Place the MovieClip suit in its layer and position it so that it covers the remodel's suit and the texture map. Give the suit the same instance name as its layer.

In the AS layer add the following script:

```
this.suit._alpha=50;
```

This sets the texture map to show through the suit, to give the gradient effects.



2 Create a symbol in your library and call this Chit. Make sure it's capitalised in the same way as the class file Chit.as on the cover disc. In the Property dialogue box, set the Linkage Identifier as Chit and the Class as Chit. (Do not use the '.as' in

the class name.) Deselect 'Export in first frame.'

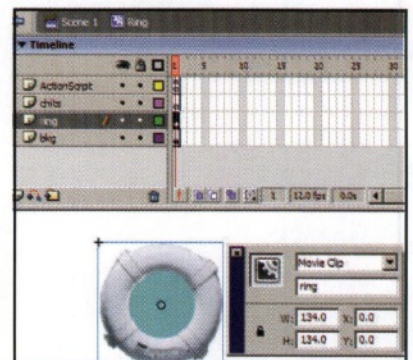
As described in Part 1, create two layers in Chit. Label the top layer as ActionScript and the bottom layer as Media. In the ActionScript keyframe place a stop();. In the Media layer place a keyframe in the second frame.

Open the library of the completed tutorial and drag the MovieClip's box and frame onto the second keyframe of the Media layer. The location doesn't matter. Open the properties and make sure that the linkage matches the MovieClip name in the library and that neither has a class attached.

Open a new MovieClip and name it Ring. You should set up the properties as described in the theory section (Ring for name, Ring for linkage and Ring for class). You can drag the Ring/Media MovieClip out of the completed tutorial onto the stage. Delete it from the stage – you should find it in the library.

You should now have the stage and two potential components ready for code!

3 Set up the Ring MovieClip as illustrated below. Place a 'stop();' script in the AS layer. You need to create a backdrop for the ring so that the user doesn't click through to the stage. Use any graphic shape you like, as long as it fills the centre of the ring.





Part 3: Writing the component code into the templates

Now your components are ready for adding all the necessary code...

Chit.as	Ring.as
Component Template Layout.	
Variable and media declaration	Variable and media declaration
Static	Static
Private	Private
Public	Public
Constructors	Constructors
Init	Init
createChildren	createChildren
draw	draw
Size	Size
Your methods	Your methods
Private	Private
Public	Public
Getters	Getters
Setters	Setters

1 Make sure that the Chit.as and the Ring.as are in the same directory as the Flash movie you laid out in Part 2. Open the templates (if you're not using Flash MX 2004, you'll need an external editor). We'll only be using the areas of the templates marked, and we'll begin our coding with the Chit.as file.

```
static var className:String = "Chit";
static var symbolName:String = "Chit";
static var developer:String = "David Aaron";
static var objectives:String = "This is a colour chit to embed into another component";
static var publicMethods:String = "flip and flop";
static var version:String = "0.9.0.1";

private var frameColour:Color;
private var hFrameColour:Color;
private var boxColour:Color;
private var boundingRec:MovieClip;
private var box:MovieClip;
private var frame:MovieClip;
private var boxHex2Base10:Number;
private var frameHex2Base10:Number;
private var frameHighlightHex2Base10:Number;
private var frameStartColour:Number;

[Inspectable(defaultValue="0xFF0000")]
public var pboxColour:String;
[Inspectable(defaultValue="0x999999")]
public var pframeColour:String;
[Inspectable(defaultValue="0x333333")]
public var pframeHighlightColour:String;
```

2 Each area of the code has a specific purpose. The execution order is init --> createChildren --> constructor (this is like code on your timeline) --> draw --> size. We declare our variables and media before our methods so that some values can be populated before we need them. This is what inspectable does for us in Flash MX 2004 Professional when it populates the Properties dialogue box.

We usually do our variables in the order of static, private followed by public. Public variables can be read by anyone anywhere in the Flash context. Our variable declaration should look like this. This comes just after the class declaration 'class Ring extends mx.core.UIComponent {'.

3 Static, private and public are colour coded (as shown above). When you enter the variables

into the template, make sure the types match the part following the ':' shown here. You can customise the default colours in the inspectables to your own colour preferences.

4 You're now ready to turn the MovieClip into a component. Right-click on the MovieClip symbol Chit in the library and select Component Definition... When the Component Definition dialogue box appears, type in the class name Chit and hit Enter. No other changes are necessary.

5 You can now go to Scene 1 and drag on a copy of Chit from the library (notice the new icon!). Look at the Properties dialogue box (seen along the bottom of this page) and you should see the three variables we marked as inspectable in our

```
function createChildren(Void):Void {
    //Set the events
    this.box.onRollOver = function() {
        frameColour = new Color(_parent.frame);
        frameColour.setRGB(_parent.frameHighlightHex2Base10);
        delete frameColour;
    };
    this.box.onRollOut = function() {
        hFrameColour = new Color(_parent.frame);
        hFrameColour.setRGB(_parent.frameHex2Base10);
        delete hFrameColour;
    };
    this.box.onRelease = function() {
        //Set _parent _parent Master colour
        _parent._parent.masterColour = _parent.pboxColour;
    };
}
```




```
function init(Void):Void {
    //Hide boundingRec and make sure it cannot receive any events
    boundingRec._visible = false;
    boundingRec._width = 0;
    boundingRec._height = 0;
    // Attach box and frame --> Do not worry about the location
    attachMovie("box", "box", 1);
    // See Note 4 about the depth settings
    attachMovie("frame", "frame", 2);
    //Note 5 Hex and based ten numbers
    boxHex2Base10 = Number(pboxColour);
    frameHex2Base10 = Number(pframeColour);
    pframeHighlightHex2Base10 = Number(pframeHighlightColour);
    //Set default colours
    setColour(boxHex2Base10, box);
    setColour(frameHex2Base10, frame);
    //Set box offset
    box._x += 2;
    box._y += 2;
    //frame is attached at 0,0 by default
}
```

```
//Private
private function setColour(arg1: Number, arg2: MovieClip): Void {
    boxColour = new Color(arg2);
    boxColour.setRGB(arg1);
    delete boxColour;
}

//Public
public function flip(Void): Void {
    setColour(pframeHighlightHex2Base10, frame);
}

public function flop(Void): Void {
    setColour(frameHex2Base10, frame);
}

class Ring extends mx.core.UIComponent {
    static var className:String = "Ring";
    static var symbolName:String = "Ring";
    // You can use this, _name="A Name" in inst for a fx instance name
    static var developer:String = "David Aaron";
    static var objectives:String = "This is a container component to hold other components";
    static var version:String = "0.0.0";

    private var changeColor:Color;
    private var currentColor:Color;
    private var temp:String;

    // Public String
    [Inspectable(defaultValue="")]
    public var targetClip:String;
    [Inspectable(defaultValue="0xFF3399")]
    public var masterColour:String;

    //Constructors
    function Ring():Void {
        temp = targetClip;
        this.onEnterFrame = function():Void {
            changeColor = new Color(_root[temp]);
            currentColor = Number(masterColour);
            changeColor.setRGB(currentColor);
            delete changeColor;
        };
    }

    //Required by Component architecture
    function init(Void):Void {
        if (masterColour == undefined) {
            masterColour = "0x000000";
        }
    }

    function createChildren(Void):Void {
    }

    function draw(Void):Void {
    }

    function size(Void):Void {
        super.invalidate();
    }

    //Application based methods
    //Private
    //Public
    //Getters
    //Setters
}
```

class. The remainder of the code can now be added to the Chit.as class file.

6 The yellow block hides our boundingRec (see above image). The green and aqua blocks attach our MovieClips that were in frame 2 to frame 1 so they can be seen. The purple box converts the data that was inspectable into a number format. (While ActionScript doesn't care if our numbers are base 10 or hexadecimal, the Properties dialogue box does. Any numerical data that is not in base 10 format needs to be brought in as a string and converted into a number. Think of it as a 'Feature.') We call a private method setColour twice. The method requires two arguments, a MovieClip and a number representing a colour. We will write setColour later – this applies the colours from the property box to our squares. The blue box at the bottom offsets our square by two pixels so that it appears in the middle of our frame MovieClip. MovieClips are attached at X=0 & Y=0 by default so our frame MovieClip does not need an X or Y position.

7 The createChildren method (see image on the left) is where all interactivity is controlled. The three events – onRollOver, onRollOut and onRelease – are all associated with the MovieClip box. The coding method should look very familiar to anyone who has written Flash MX code. In the onRelease event there is a special reference to the _parent._parent.MasterColour. This variable was not declared with the other variables because it will be part of the Ring component.

Experienced Flash MX 2004 coders generally do not code events like this when writing professional components. The current best practice recommended by Macromedia is to use listeners to monitor change events through the mouse or keyboard. If you run 'trace variables' under the debug menu when you're testing your code, you'll notice that Flash adds these to your code.

```
MovieClip: Target="_level0.instance1"
Variable _level0.instance1.
addEventListener = [function
    addEventListener ].
```

If you're very comfortable with ActionScript, you can make the changes, otherwise no one will know because once a component is compiled the code is hidden from spying eyes!

8 The custom methods section (see top-right image) follows the createChildren method. The private method has been called already in the init method and is used to change the colour of MovieClips. The two public methods, flip and flop, are not required, but would enable the visible rollover states to be triggered by code outside of the component.

Test your component. If it picks up your colour preferences from the Properties panel and the frame MovieClip changes colours when you move the mouse over it, you're ready to compile your clip. Right-click on your component and select 'Convert to Compiled Clip'.

9 Open the MovieClip Ring. It's time to bring Chit and Ring together. On the chits layer drag out the compiled chit components five times and arrange them tastefully – the component should now be called Chit SWF. Give the Chit components instance names of your choice. Give each one a unique hexadecimal colour in the pboxColour in the parameters panel. You can change the frames from your default if you like. The Ring code is simpler than the Chit code so it has been added for you.

The constructor code (see above) runs as though it were entering the frame with the MovieClip. While there are other ways that are more code efficient, this approach is reliable and easier to script. The constructor polls the Chit components and checks to see if any have changed and then changes the MovieClip colour on the stage. Test the component and compile it!

10 You're now ready to drag the Ring SWF onto the stage. There's nothing left for you to do now but test it. ■



Information management



All the components needed to complete this tutorial are on your cover disc.

Tom Knowles shows you how to create an online notebook and to do list in Flash and Fireworks

For a while now, Macromedia has been pushing Studio MX as the one-stop-shop for delivering Rich Internet Applications (RIAs). RIAs are Flash-based apps that work in the browser but have the feel of traditional desktop software. This means stepping away from the page-refresh paradigm and into the realm of true interaction.

Examples of RIA include booking forms, software demonstrations and product customisation applications. These sorts of

things are ideally suited to the rich Flash environment, which can provide the user with interactive GUIs and the ability to accomplish their tasks within one page and with instant feedback.

Here we're going to produce a fairly simple online information manager, which allows the user to store and retrieve their information using a text field, and also maintain an Outlook-style to-do list. We'll design and structure our interface in Fireworks and Flash, before using

ActionScript 2.0 to bring the application to life. We'll also use Flash's cookie in the SharedObject to save the user's details.

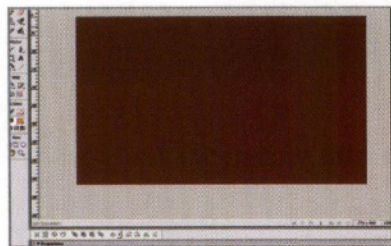
The SharedObject was initially an undocumented feature, but Macromedia has decided that it was useful enough to be worth full support. In this tutorial we only touch on what it can do but, with careful planning, it can be the basis of some powerful applications. It's even possible that it could replace the cookie as a means of storing information on the client. ■

Part 1: How to create the notebook

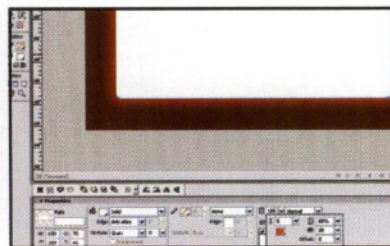
As usual when designing a Flash application layout, we'll start with Fireworks...

Shared Objects

Shared objects are a hugely useful tool, but, as with any client-side data storage, they cannot be totally relied upon due to the user's ability to remove them at any time...



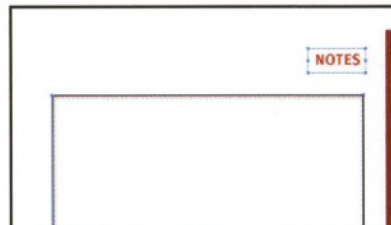
1 Open Fireworks and create a new document of around 770 by 460 pixels in size. In the canvas properties panel, select a dark red for the canvas colour, then select the rectangle tool from Vector menu.



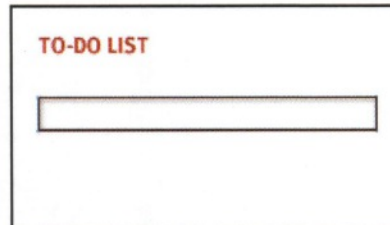
2 Draw a white rectangle of 630x320 pixels on the stage. Centre it using the Align panel (Window > Align), and with it selected, return to the properties panel. Specify a Rectangle Roundness value of 20 per cent, and in Effects, add a soft red glow.



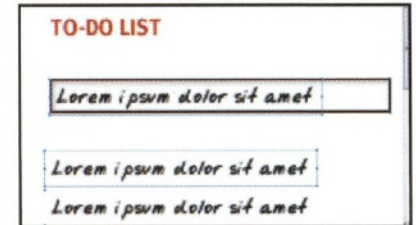
3 As above, add a title above your rectangle. We've used small white text so, to make it stand out a bit more, add a 0 pixel black drop shadow. The reason for the 0 measurement is that it causes the placement of a subtle shadow around the edges of the object.



4 Next, let's create the note-taking aspects. Use the type tool to write out notes, again using a drop shadow if you wish. Again select the rectangle tool, this time with a red outline, and draw a 270x230 box. Add a 2 pixel grey inner shadow in the effects panel

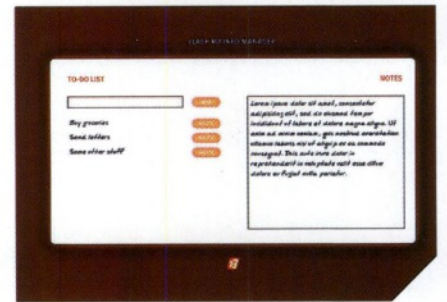


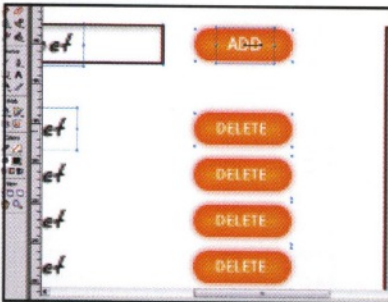
5 We now need to draw another field for the to-do list. It makes sense to copy and paste the existing note-taking rectangle and resize it, but if we just use the transform tool, the inner shadow will be affected. Use the subselection tool instead, which allows you to maintain all applied effects whilst resizing.



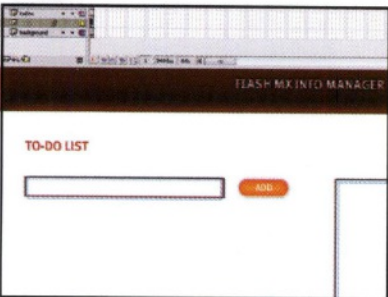
6 Handwritten fonts can be awfully gimmicky, but occasionally a reasonably subtle one can add a touch of informality to any interface. We've used one for the user-input text in our design. Be warned however: if your handwritten font is complex, it may be rendered illegible by Flash's currently primitive type engine.

THE FINAL DESIGN

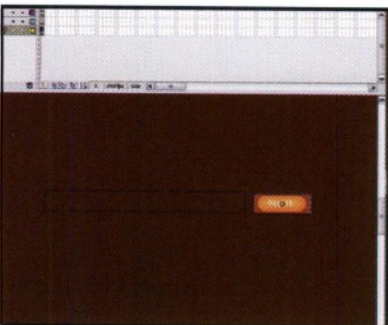




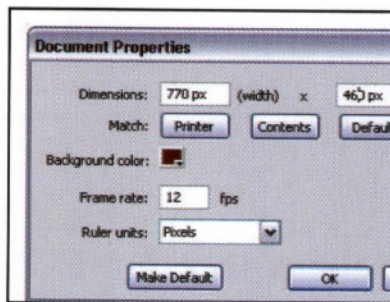
7 To create the buttons above, use the rectangle tool (with 100 per cent roundness), and fill it with the contour gradient. We've used tints of orange to make up the gradient, although this isn't strictly necessary as long as you take care to avoid clashing colours. As a final touch, add a 0 pixel drop shadow.



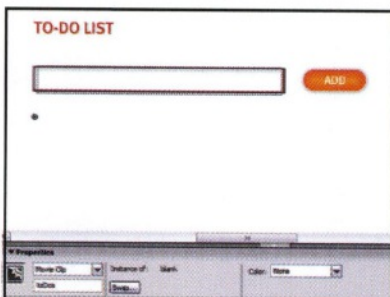
10 Move to the fields layer, and create two new input textfields, using the handwritten font you used in your original Fireworks design. Name the fields `todoField` and `notesField`, and ensure the font is embedded for both (click Character in the properties panel), and that `notesField` is set to multiline.



13 On the first layer, type a `stop()` action. Next, create a dynamic textfield, as above, with the instance name `itemtxt`. Then copy and paste the delete button we drew in Fireworks. Above it, create a white rectangle with alpha value 30. Convert the rectangle to a MovieClip and give it the instance name `delbtn`.



8 That's all there is to designing our interface. Open Flash MX 2004, and create a new document of 770x460 pixels, before returning to Fireworks. For Flash environment, we aren't going to need the handwritten text we've used, so delete that for now, along with the delete buttons. Select all (Select > Select All), and copy.



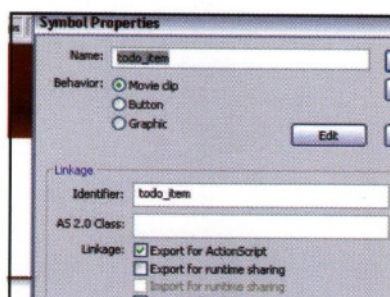
11 On the `todos` layer, choose Insert > New Symbol > MovieClip. Don't add anything to the new MovieClip; instead return to the main time line and drop this blank clip onto the stage, giving it the instance name `todos`. In our ActionScript code later on, we'll use this to attach to-do items.



14 Back on the main time line – on the `addbtn` layer, drop another instance of your button MovieClip over the add button graphic. Give this clip the instance name `addbtn`, and then move to the actions layer. If you have not already done so, open the Actions panel (Window > Development Panels > Actions).



9 Jump back into Flash, and move to the time line area. Rename the first layer functions, and then create five more layers as above, named actions, todos, field, and background. On the background layer, paste the Fireworks objects we copied earlier. As you'll see, Flash automatically converts them to a MovieClip.



12 Again, create a new MovieClip, and call it `todo_item`. Whilst still in the Symbol Properties dialogue box, click the advanced tab and Export for ActionScript. Give the clip a linkage identifier of `todo_item` as well, and check the Export in first frame box. Now create two new layers.



15 Type in the above code snippet. Other than the obvious `stop()` action, all this does is call the `init()` function (which we'll write later on). This function serves as an initialisation procedure for retrieving saved data and formatting the interface accordingly. Finally, move to the functions layer to begin writing our main code.

Avoid obstruction

If your aim is to provide a truly useful tool, you'll need to avoid obstructing your users' workflow with needless animations and graphical fripperies and, instead, focus on providing your functionality in as smooth a fashion as possible.



Part 2: Actionscript code part 1

Arrays are useful things...

Arrays are great for storage: whether for lists of global state variables or for something as low-key as helping the manipulation of strings. In the following code, we'll make good use of them to store information in the Flash cookie (the SharedObject), as well as to maintain data within the functions themselves.

Arrays' only problem is that they don't come with enough methods included. The reason for this is that their use is so wide-ranging (and often so esoteric) that the number of potentially useful methods is almost infinite. Hence they come with basic building-block methods such as push(), pop() and splice(). So when we need a specific method that the Array doesn't have out-of-the-box, we just have to write one and plug it in via Prototype syntax. In fact, extending the Array object is the first thing we do in the first part of our code.

1 Lines 1–14: Here is the Array prototype function we mentioned earlier on. Called deleteAt, it allows you to remove an element (specified by the index parameter) from any Array, without having to manually resize it, or pass it a range of values (as with the splice method). Notice how it builds on various existing Array methods; if we want to remove the first element, we just use the shift() method. If we want to remove the last element, we use the pop() method. Any other element, and we use splice(), which removes a given range of elements.

```
Array.prototype.deleteAt
= function(index)
{
    if(index == 0)
    {
        this.shift();
    }
    else if(index ==
        (this.length-1)){
        this.pop();
    }
    else if(index > 0 &&
        index < (this.length-1) )
    {
        this.splice(index,1);
    }
};
```

2 Lines 15–19: The init() function essentially just initialises the application. At the beginning, we retrieve the notes data, and then initialise the Notes and the To-Dos. Notice how we're layering our application; by splitting up the code into small pieces we can structure it more logically, and make it much easier to maintain and edit.

```
function init()
{
    retrieveNotesData();
    initNotes();
    initToDos();
}
```

3 Lines 20–33: Here we deal with the add button's rollover and rollout functionality. We've kept it extremely simple: when the user moves their mouse over the button, its _alpha value becomes 0 (remember that the button in question is actually just a white rectangle laid over the actual button image). When the user rolls out, we set the _alpha value back to 100. When the user actually clicks the button

– ie releases the mouse button over it – we call the addToDo function, passing it the text currently in the toDoField textfield. Again, we're breaking the code down into small pieces; we could place the addToDo functionality here rather than in a separate function, but it would only make things messy.

```
addbtn.onRollOver
= function()
{
    this._alpha = 0;
}
addbtn.onRollOut
= function()
{
    this._alpha = 100;
}
addbtn.onRelease
= function()
{
    addToDo(toDoField.text);
    toDoField.text = "";
}
```

4 Lines 34–45: The initToDos function. We declare a couple of global variables here – the _global keyword means that the variables are accessible from anywhere in the movie. This is very convenient, but if you want to do this you need to be certain that the variables in question have unique names. Otherwise, you could be in serious danger of variable conflict and difficult-to-solve bugs.

The most important variable here is the toDoArray. This is the Array that we're going to use to store all our to-do information, as we pass it around the movie. Also notice that we use the SharedObject here. Its syntax is reasonably straightforward: we assign it to a variable and give it an identifier. Finally, we check to see if we've got any previously stored data. If we have, we stuff it into our toDoArray and call the arrangeToDos function.

```
function initToDos()
{
    _global.itr = 0;
    _global.toDoArray =
        new Array();
    _global.gap = 25;
    var my_so = SharedObject.
        getLocal("info");
    if (my_so.data.toDoArray
        != undefined)
```

```
{
    toDoArray =
        my_so.data.toDoArray;
    arrangeToDos();
}
```

5 Lines 46–54: The addToDo function creates a generic object called toDoObj, and stuffs it with the txt variable passed to the function. We then push the object onto the global toDoArray. Why use an object? Well, we could just stuff the Array with String variables, but it makes much more sense to use objects, because they can store more than one item. In fact, they can store as many as you want them to. Suppose we wanted to allow the user to store dates and times in their to-do list, by using an Array of objects, we don't have to make as many changes to the code, because the Array doesn't care about what is in the objects it's storing.

```
function addToDo(txt)
{
    var toDoObj = new Object;
    toDoObj.txt = txt;
    toDoArray.push(toDoObj);
    var my_so = SharedObject.
        getLocal("info");
    my_so.data.toDoArray
        = toDoArray;
    arrangeToDos();
}
```


Part 3: Actionscript code part 2

Arranging the to-do list...

This part of the code deals predominantly with arranging the to do items; a particularly noteworthy aspect being the laying out of items when the user has deleted one. Another area to pay attention to is the use of the SharedObject. You'll see that we continually update it, but never explicitly save it. This is because the Flash player automatically saves all open SharedObjects when it's closed, thus saving us the work of collating and storing the data.

Lastly, note the use of the var prefix whenever we declare variables. This localises the variable within the current scope; for example, the function or the object it was declared in. The reason for doing this is to keep things more organised; preventing variables with the same name in other scopes causing interference and potential conflict. It might seem a little cumbersome at first, but it's worth getting used to if you want to avoid irritating and hard-to-find bugs.

1 Lines 1–12: The arrangeToDos function is the meaty heart of our code. The first thing we do is create a couple of variables, a local y value and an Array for use in the layout. Next, it's time for a classic For loop. Basically, we're going to loop through the global toDoArray (we don't need to use any prefixes, we can just access it from anywhere), and for each item we'll attach an instance of the todo_item MovieClip. See how we create a reference to the current clip via the eval command, and then retrieve the toDoArray's current object's values. A quick tip worth remembering is to always give a MovieClip or object in a For loop an index variable, based on the iterator used in the loop. This is incredibly useful for when the MovieClip/object wants to identify itself later on. Rather than parsing its actual name as a String, or looping through all its associated objects, you can simply use the index value. We don't make a huge amount of use of this functionality in this example, but it's worth getting into the habit regardless.

```
function arrangeToDos()
{
    var y = 0;
    clipArray = new Array();
    for(var i=0;
    i<toDoArray.length; i++)
    {
        toDos.attachMovie
        ("todo_item", "todo
        _item"+i, i);
        var temp = eval
        ("toDos.todo_item"+i);
        clipArray.push(temp);
        temp.itemtxt.text
        = toDoArray[i].txt;
        temp.index = i;
        temp._y = y;
```

2 Lines 13–20: As before, we use the overlaid delbtn MovieClip as a button, and manipulate its _alpha values for the onRollOver and onRollOut events.

```
temp.delbtn.onRollOver
= function()
{
    this._alpha = 0;
}
temp.delbtn.onRollOut
= function()
{
    this._alpha = 100;
}
```

3 Lines 21–35: The onRelease event is called when the user clicks the to-do item's delete button. The first thing we is to remove the corresponding object from the toDoArray. How do we do this? Via our clever little index variable. If we hadn't stored this value in our initial loop, we'd have to loop through all the Array's items until we found the one whose data matched the item we'd clicked. And that's not perfect either because it's feasible that there might be more than one item with the same data. After this, we use a quick For loop to remove all the current to-do items from the screen, then update our SharedObject with the newly-trimmed toDoArray. Finally, we make a call to the arrangeToDos function again.

```
temp.delbtn.onRelease
= function()
{
    toDoArray.
    deleteAt(this.
    _parent.index);
    for(var i=0;
    i<clipArray.
    length; i++)
    {
        trace(i+deleting i +
        clipArray[i]);
        removeMovieClip
        (clipArray[i]);
    }
    var my_so =
    SharedObject.
    getLocal("info");
    my_so.data.toDoArray
    = toDoArray;
    arrangeToDos();
}
y+= gap;
}
```

4 Lines 36–45: These initialisation functions are fairly simple. initNotes creates an onChanged handler for the notesField textfield. This is triggered, unsurprisingly, every time the field is changed and fires off the storeNotesData function. What does this do? Simple really: it opens the info SharedObject and updates its notes variable, as well as taking the opportunity to update the toDoArray.

```
function initNotes()
{
```

```
    notesField.onChanged
    = storeNotesData;
}
function storeNotesData()
{
    var my_so = SharedObject.
    getLocal("info");
    my_so.data.notes
    = notesField.text;
    my_so.data.toDoArray
    = toDoArray;
}
```

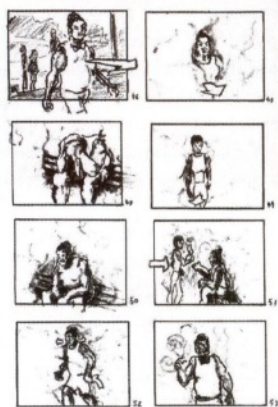
5 Lines 46–53: The final function, retrieveNotesData, does exactly what it says on the tin. It creates a reference to our SharedObject, and then checks to see if the notes variable actually exists. If it does, it must mean that we've previously stored some data inside, so we grab it and pump it into the notesField textfield out on the Stage.

That's all there is to it! By breaking down a reasonably complex problem into the smallest constituent parts possible, we've made the code both easier to understand as well as easier to modify and optimise. When writing the first iteration of any application, it's always tempting to bung all of your code into one big function. This might be quicker, but in the long run it always means headaches.

```
function retrieveNotesData()
{
    var my_so = SharedObject.
    getLocal("info");
    if (my_so.data.notes
    != undefined)
    {
        notesField.text =
        my_so.data.notes;
    }
}
```


GET STARTED WITH...

Storyboarding



A pre-production storyboard for short film *The Assassins' Opera*, by freelance artist Lee Andrews. Andrews used crushed charcoal, pen and ink – an idea “cribbed from an old book” he found – to represent the steamy sauna setting for the action.



All the components needed to complete this tutorial can be found on the cover disc.

Learn more online:

www.colonymedia.co.uk
www.hiveuk.co.uk
www.idmedia.com
www.jellyfishpictures.co.uk
www.leeandrews.co.uk
www.storyboards.co.uk
www.thebestthingsince.com

Whether you're destined to be the next Martin Scorsese or just planning out a simple web site animation, preparing a good storyboard is often the best plan of action. **Graeme Aymer** finds out why

Col. John “Hannibal” Smith was right. As leader of the A-Team, he was always pleased when a plan came together. But you don't need to lead a band of renegade Vietnam veterans, found guilty of a crime they didn't commit, to share that sense of satisfaction. Bringing your artistic visions and planning together in the form of a storyboard will enable you to clearly communicate your ideas to colleagues or clients and ensure success.

So, back to basics. A storyboard is essentially a plan, a visual sketch that sequentially depicts action, scenery characters and movement within movies, live action or animation. However, storyboards are alive and kicking on the small screen, too. They're a staple of the advertising industry, again for live action and effects-driven pieces, channel identities and bumpers. Beyond that, they help interactive designers plot Flash animation, plan interactivity and user experience for web sites and DVD menus, and even help to win client pitches.

But what constitutes a storyboard? “Depending on the size of the project, it could be just a simple set of flat screens, or it might just be a sketch on a piece of

paper,” explains Laura Jordan, head of design at I-D Media.

Scribbles

“I've seen jobs won on the basis of a scribble on a piece of paper,” adds Jason Keely, a freelance storyboarder and visualiser, Hive associate and ex-Lambie-Nairn staffer.

“We tend to use our boards with the

of what it's going to be like.”

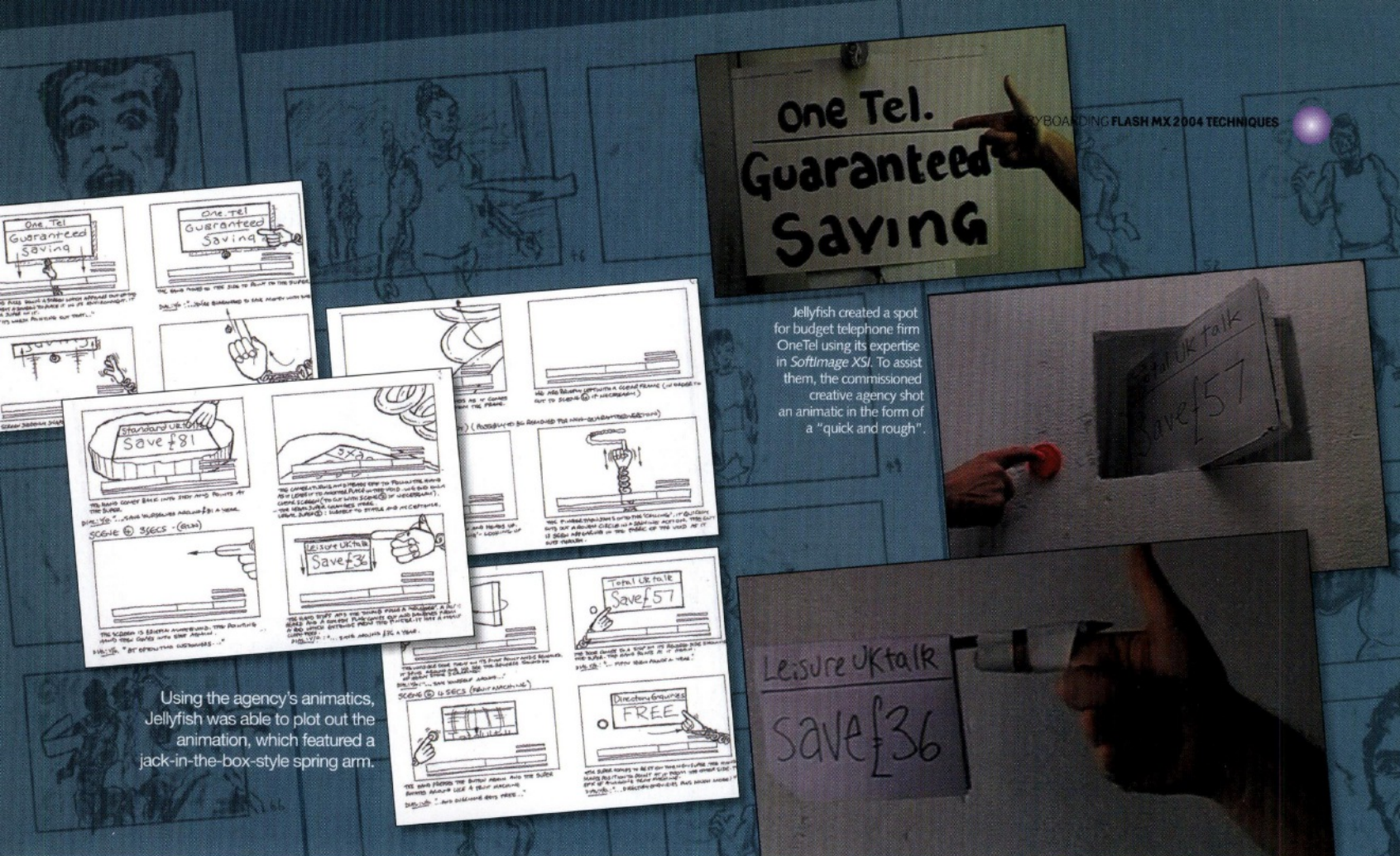
Then there's that peculiar cousin of storyboarding: animatics. Typically, this is a rough animation based on scanned storyboard frames and then edited to a dialogue and soundtrack in order to test a concept for creatives or even for market research. Related to this are “Stealomatics” (animatics made up of “stolen” footage and audio, put together to

“It can be as simple or as sophisticated as you want... black and white line drawings or a highly-finished quick and rough”

same techniques we're going to use to do the job,” contrasts Will Rockall, director of Softimage XSI specialist facility, Jellyfish. “If it's going to involve 3D animation, we might make elements up in 3D to put in the board. But if it's more of a composited 2D type of thing, we would probably use Photoshop and After Effects to create the board. We don't usually draw with pencil; we try to work them up so we get an idea

explore potential mood or tone for a project), “Photomatics” (animatics using transitioning still photography) and “quick and roughs” (a version of a commercial, shot on a low budget using the script). Animatics are usually the buffer between the storyboard and final shooting stage.

“It can be as simple or as sophisticated as you want,” says Ralph White, MD of storyboarding agency, Storyboards. “You



one Tel.
Guaranteed
Saving

Jellyfish created a spot for budget telephone firm OneTel using its expertise in *Softimage XSI*. To assist them, the commissioned creative agency shot an animatic in the form of a "quick and rough".

Using the agency's animatics, Jellyfish was able to plot out the animation, which featured a jack-in-the-box-style spring arm.

STORYBOARDING FAVOURITES

When it comes to storyboarding, pen, pencil and paper remain the most commonly-used tools. Software-wise, Photoshop, Illustrator and Freehand are fine for scanning and colouring, and Final Cut and After Effects remain popular for animatics. 3D is created using anything from Softimage to Cinema 4D. The following dedicated apps could also prove useful.

Boardmaster

(www.boardmastersoftware.com)

The ability to import your own artwork, an easy-to-use animatic function and free clip art mark out this PC-only storyboarding app. (\$45)

Frameforge 3D Studio

(www.frameforge3d.com)

With Frameforge's virtual 3D world, you can set up scenes with precise camera angles, posable characters and 3D objects. (\$349, plus shipping)

Realviz StoryViz

(www.realviz.com)

A pre-visualisation tool for setting up rooms, placing actors and then simulating where you need cameras in the 3D space. Windows 2000/XP only. (3,000 euros, approx £2,050)

Sketchbook Pro

(www.alias.com)

As the name suggests, this software enables you to doodle and draw

directly onto your tablet PC's screen, or now on a Mac OS X-driven Wacom tablet. (£135 excl VAT; or free trial)

Storyboard Artist

(www.storyboardartist.com)

A more fully-featured version of *Storyboard Quick*, *Storyboard Artist* enables motion graphics. Both apps include fields for scripts and comments. (\$799.99)

Storyboard Lite

(www.zebradevelopment.com)

Another 3D pre-visualiser. Add your own background, manipulate the included props and characters and test camera angles. For Mac and PC; there's even a Linux version. (\$500)

Storyboard Quick

(www.storyboardquick.com)

Features character, prop and scenery libraries. You can even integrate your photos with its clip art. Well featured, but at \$279.99 you'll need to draw well.

can do an animatic with black and white line drawings, quite literally, or you can do it with highly-finished quick and rough with all the actors, which depends on how you want it to be researched and what they think is going to be the best way of getting the idea across."

Storyboarding enables I-D Media to map out potential user journeys — to map out visually the experience of downloading a recipe or making an online purchase. "Using a storyboard is absolutely vital for everything we do," says Jordan. "Without them you can't be sure of the user experience."

Not only does this enable the company to plot user experience and check visuals against each other, it also serves as a kind of inventory, to ensure nothing is left out. In addition, it means that the design team can check to see that all the elements required are included.

For I-D Media, however, storyboards aren't just about organisation — there's a real financial consideration. "It also gives the client something physical that they can sign off," says Jordan. "So if they decide they want something that isn't actually what we've done, we can charge out of scope for that, rather than tweaking

all the time. That's where you lose a lot of money on a project, that final ten per cent of moving things around."

Sliced bread

Christina Villics and Jamie Denham, co-directors of animation studio Sliced Bread, use a more text-based site map method when it comes to their more static sites. However, when it comes to sites featuring motion graphics, they're not averse to the use of storyboards, to work out both interactivity and straightforward Flash animation.

"I did a storyboard for our web site because it's very visual," explains Christina Villics. "It's not the usual 'press a button', and you always have animation. The programmer needed a storyboard to see how it followed up when you pressed a portfolio button and the toast came out of the toaster. There's then another menu on the slice, which comes later, and you have a 'back to menu' button, so he needed to see what happened with the animation."

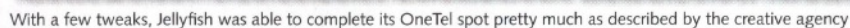
As for what makes a good storyboard, the jury is out. They should feature sequential frames of key shots, including directional arrows to denote movement or

DO:

- Learn to draw – you'll need this skill if you want to be a pro.
- Brush up on your film grammar – watch as many flicks as you can.
- Have a concept. A slick storyboard can't hide a poorly conceived idea.
- Sign up to Shooting People (www.shootingpeople.com) for help.
- Get into animation. Watch classic cartoons (Disney, Hannah Barbera or Warner Brothers).
- Use all the tools at your disposal. For live action, pen and paper might be best; for the abstract, perhaps Photoshop or Flash.
- Make a storyboard for interactive projects – it'll help you check that everything works as intended.
- Learn from other people's work.
- Keep a sketchbook. Be inspired anywhere. Challenge yourself.
- Be patient. Having to make 100 sketches of your site or film will pay dividends in the end.

DON'T:

- Be afraid of detail. If you're trying to secure a pitch or funding, the more your backers see, the better.
- Be afraid to improvise – it may add spontaneity and life to your work.
- Worry if you're not a brilliant drawer. Do the best you can.
- Forget to number your frames and pages. Reference points for your crew are essential.
- Go with the obvious solution. Before you get cracking, think about other approaches.
- Get too wrapped up in your own idea. Remember, this can be a collaborative venture.
- Be obsessed with technology.
- Be too slick. Your boards need to express ideas in an inviting way.
- Forget the written word. Notes or snippets of script can caption your pictures and help your colleagues.
- Be afraid to experiment. Using different software or ink could completely transform your board.



actions essential to the plot. But the debate rages as to detail. "As long as it gets the idea over, the concept, and the actual look of the final piece, then I think it could be as long or as brief as needed," says Jellyfish's Will Rockall.

"It's about communication," explains Keely. "How that's done doesn't matter. The thing is to be clear about what you're setting out to do. It should either contain very, very honed down and accurate visuals, or very loose drawings with some poignant reference points.

Either way, it's all about communicating as clearly as you can. Martin Lambie-Nairn taught me that when you're communicating your ideas, you should communicate them as if you're talking to an eight-year-old. Don't talk to your clients as if they're stupid, but be clear

about what it is you're trying to say. The storyboard helps with that."

"It's like reading a book: it takes your idea right through," says Denham, "and that works for interaction as well. When we're creating DVDs, we can make sure of what goes in there. I find I often come across problems when I'm creating interaction where you see a link that doesn't quite work."

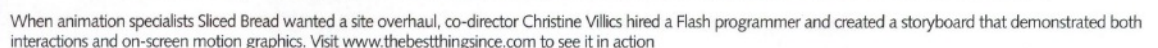
Pen vs machine

Industry professionals tend to favour good old-fashioned pen and paper over dedicated storyboarding software. "It immediately suggests a bunch of templates. It isn't something we've dreamed of using," says Jason Keely. This, of course, suggests that the best storyboarders can always draw well.

"You have to be able to see time," explains Keely. "So in two seconds you have to know what's going to be where. You have to have a feeling, not just about how it's going to look, but also how it's going to be over time."

"When all is said and done, you've still got to be able to draw," says White. "If you can't draw, then you're in trouble. A good storyboard artist must also be aware of what the script is trying to achieve, because that helps in the characterisation of people."

"I've got books with Martin Scorsese's drawings for the big shootout sequence in *Taxi Driver*," counters filmmaker and director of Colony Media, Mario Cavalli. "Scorsese can hardly draw, but the storyboard is extremely specific and totally functional. You can look at it and



THE OLYMPICS, BBC STYLE

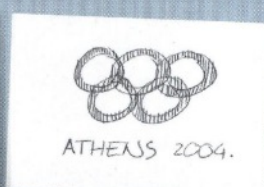
Storyboards was asked by agency DFGW earlier this year to provide artwork to promote the BBC's Olympics coverage. Ordinarily, Storyboards is approached by an agency to produce an animatic based on a brief or script.

A director is then chosen to bring the boards to life. In this case, however, the

director had already been chosen and he'd started on traces, or sketches, for the planned action. These were scanned and fleshed out by Graham Barratt and Storyboards using standard image-manipulation software.

Storyboards then set out to produce the animatic, working out the particulars of

shot timings in an edit suite at sister company Locomotion Creative Productions, before the piece was finally shot. The agency producer was Sally Fisher, the agency creative, Simon Riley, Graham Barratt the storyboards producer and Passion Pictures the production company.



DFGW chose a director who had already created traces of key scenes. It was then up to Storyboards to create colour boards and appropriate animatics.

Storyboards' MD Ralph White is adamant that storyboard artists should be able to draw. While detail may not be important for the boards, it's certainly worth getting your pen and pencil chops together.

In some cases, traces, animatic and final shots are almost a perfect match.

To add detail to storyboards and animatics, Storyboards uses nothing more complex than the standard Adobe tools – Photoshop and Illustrator. The company looked into storyboarding software, but found the template-driven approach inadequate.

“Martin Scorsese can hardly draw, but the [*Taxi Driver*] storyboard is extremely specific and totally functional”

know exactly what that sequence is going to look like. It helps if you can draw well, but it's not essential. The most important thing is that it conveys the correct information.”

“I think it's something you do learn as a storyboard artist – you have to be able to put expression and character quite quickly and instantly into your creations,” adds freelance storyboard artist Lee Andrews.

“I think the crucial thing with storyboards is that you work very fast and you work a lot. It's not as important to have a style as it is to have a way of communicating things very quickly and easily.”

Expression

“To my mind, it's better to work roughly and expressively than be very polished and not get that [expression] across,” he continues.

“It's all very much about storytelling: you have to be able to tell your story in the pictures you create.”

Essentially, storyboards are about ideas. Without ideas, you have no chance. Whether you're working on a corporate web site or the next Pixar animation, you'll need a central concept, and in order to communicate with your team, or even just to keep the strands of your personal portfolio site together, you'll need a plan.

If your hope is to join Ralph White's band of pros, you'll need serious drawing chops, but Scorsese's crew was perfectly able to follow his scrawled instruction, according to Mario Cavalli.

So whatever you do, have a plan, have a good storyboard and take great pride when it all comes together. ■

DESIGNER TIPS



“Have lots of patience, be prepared to work for free for a while just to get into it,

and then keep your eye out for opportunities.” – Lee Andrews, freelance storyboard artist. (Lee got his first big breaks from filmmakers' community site Shooting People.)



“Watch really bad American TV, the *Dallas* type, but with the sound turned off,

so you're just looking at the shots, the composition and the camera moves. In terms of the fundamental elements of film grammar, you can learn the basics in an afternoon.”

– Mario Cavalli, filmmaker and director of Colony Media

FURTHER READING...

Film Directing Shot by Shot:

Visualizing from Concept to Screen (Steven Katz)

Michael Wiese Productions
ISBN 0-94118-810-8, £25

The pros rave over this one – and there's a terrific section dedicated to the storyboarding art. Highly recommended.

Filmmaking for Dummies

(Bryan May Stoller)

John Wiley & Sons

ISBN 0-76452-476-3, £13.95

Provides good, basic coverage.

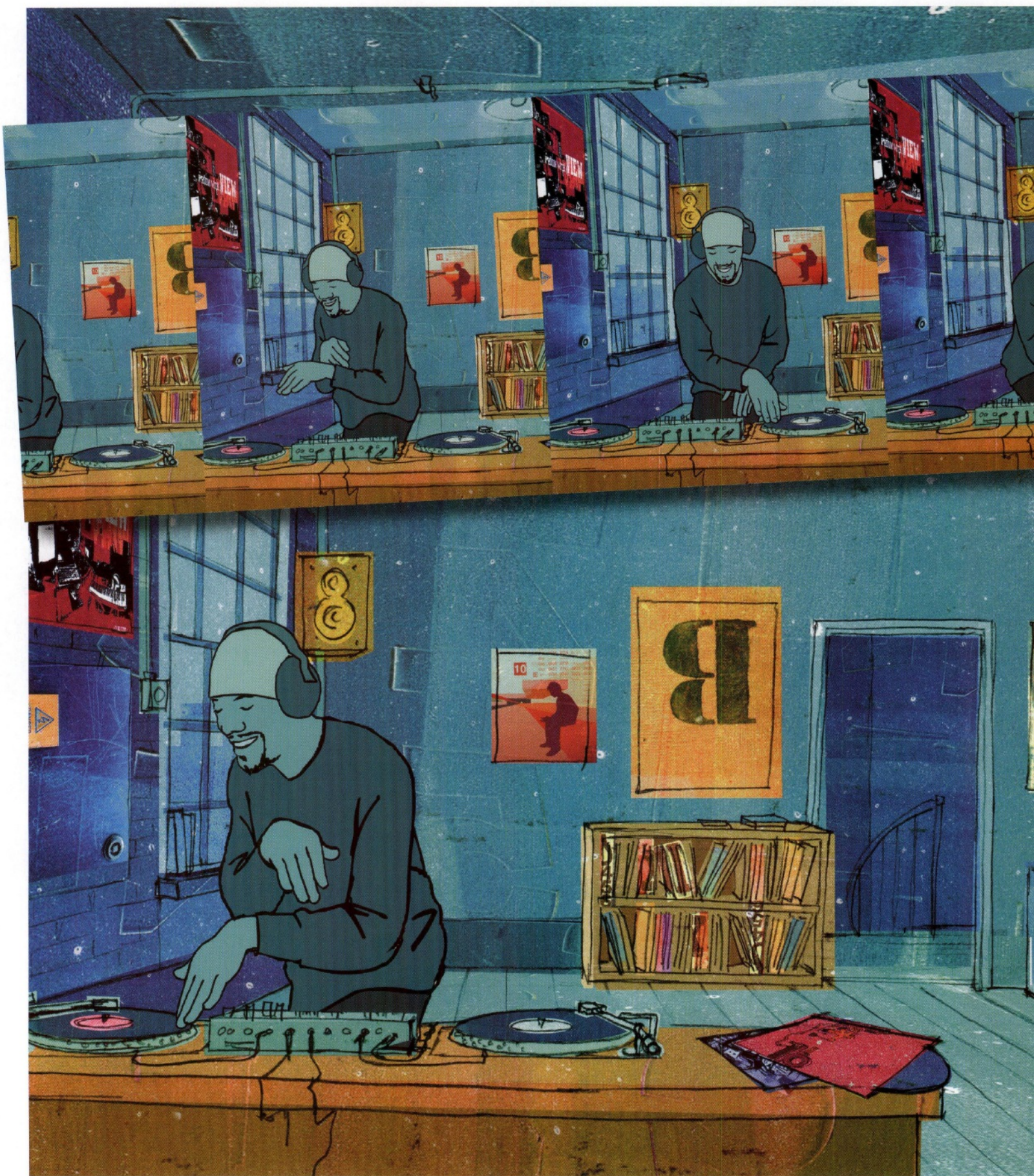
The Animator's Survival Kit

(Richard Williams)

Faber and Faber

ISBN 0-57120-228-4, \$30

For animation, character movement and character design, this is the best book in town, penned by the man who brought Roger Rabbit to life.



Rotoscoping Video

By cleverly combining Flash MX video with old-school drawing techniques, you can create your own energetic animated movies for broadcast or the web, as this award-winning piece by **Luke Whittaker** demonstrates

Starting out as a simple animation tool to counter the growth of the animated GIF, Flash has since evolved into a powerful piece of software for designers seeking to make their creations move. But its impact on the Web has resulted in hand-drawn techniques often taking a back seat to tweens, and a "sleeker", less figurative style of animation.

It doesn't have to be like that, of course. In this tutorial, I'm going to show you how it's possible to recapture that more traditional look – using Flash and a little elbow grease. Ideally, you'll need a graphics tablet and a camera to capture the moving images, but after that it's up to you.

With Flash MX, the ability to import movie files directly into the timeline makes it possible to perform what's called "rotoscoping" – where you "trace" the real-life footage to create highly fluid, naturalistic animations. Even if you feel you can't draw, Flash makes it much easier to create exciting and individual material.

In this tutorial, I'll illustrate how to prepare your video in Premiere, then rotoscope the footage in Flash. Afterwards, you'll combine the animation with an image created in Photoshop, using hand-drawn elements to give life to the surroundings. Pinpoint accuracy isn't the key here; the goal is to develop a coherent, unique and eye-catching piece of work.

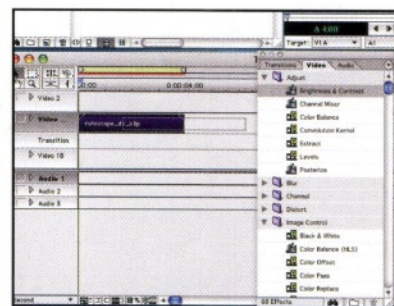
A completed version of the animation is provided on the cover disc for reference. ■

Video: Rotoscoping

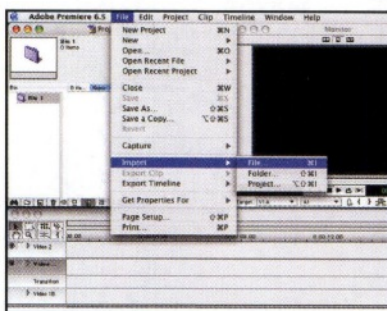
Getting a traceable image...



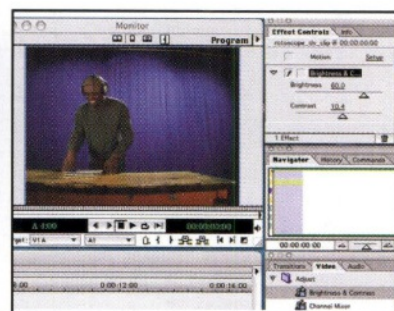
1 Before you start, sketch a plan of how you want to compose your shot. This will help you decide from what angle to shoot your footage, and also how to organise your background image. Here, I've decided to film my character head-on, which enables me to drop in a background that adds a lot of depth.



2 For this tutorial, I'm using Premiere, but other DV editing programs will do just as well. Open Premiere and import the file *Rotoscope_dv.mov* from the cover CD into the Bin by choosing File > Import > File. If you're using your own footage, import using File > Capture > Movie Capture.



3 You now need to adjust the image to bring out the qualities required for tracing. Drag the *rotoscope_dv* clip to the timeline and, within the Video Effects tab, select the Adjust dropdown.



4 Within Adjust, drag Brightness & Contrast onto the clip and adjust with the Effects Controls until you feel the image is bright enough to draw over in black. The overall look isn't important. You only need to ensure that the image is easily traceable. ➤



Expertise provided by Luke Whittaker
You can contact Luke at
www.lukewhittaker.com



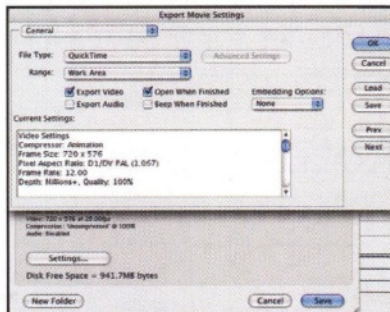
Files on disc
You'll find the files you need on the cover disc.



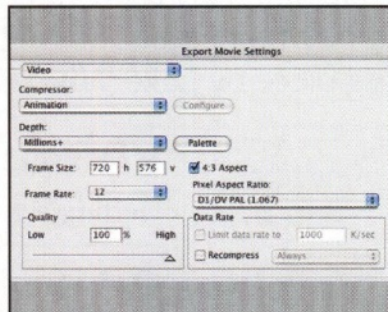
Video: Rotoscoping continued

It's all in the detail

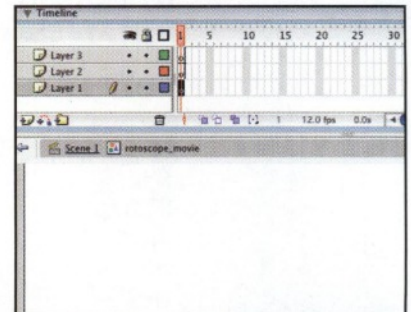
If you're keen on creating detailed facial expressions or a more intricate style of animation, it's important to use a high-quality medium like DV over a digital camera, unless you're up close. Set your shot up in front of a flat-colour background – in this case, I've used a blue-screen in a studio. It'll take up a little more time (and, for a while, a little more hard drive space), but when rotoscoping you'll appreciate not having to guess where your character's nose is.



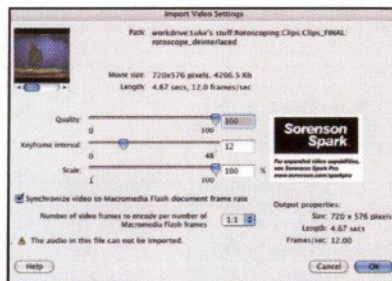
5 Now export the clip by choosing File > Export Timeline > Movie. Make sure File Type is set to QuickTime and un-tick Export audio. Click Settings and under the tab labelled General, select Video.



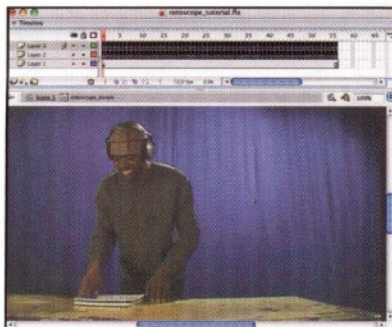
6 Change the frame rate to 12fps (enough to keep the movement fluid) and set the compressor to Animation (this will result in a large file size, but it's quicker to export and image quality is important here). Under Special Processing, click Modify and tick the box labelled De-interlace to make sure the image is crisp. Name your clip "Rotoscope_deinterlaced.mov" and click OK to export it.



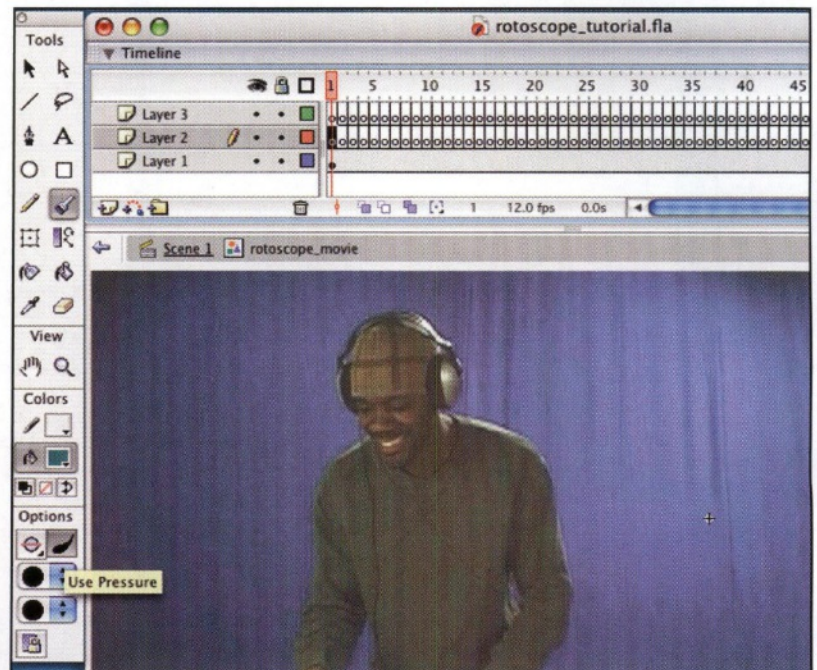
7 Open Flash MX and set the Frame Rate to 12fps. Create a new Graphic Symbol (Insert > New Symbol). Create three layers within it and select the lowest one.



8 Click File > Import and locate your exported movie. Select Embed Video and, in the box that appears, make sure the box "Synchronise video to document Frame Rate" is ticked. Quality should be set to 100.



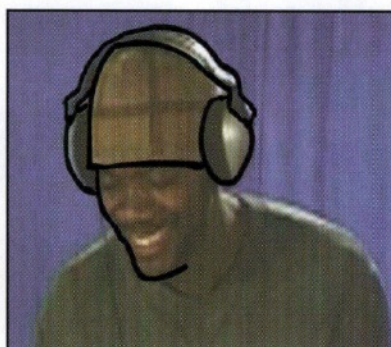
9 When asked if you want to insert more frames into your movie, click Yes. Lock the bottom layer where the video now sits. This is the guide layer you'll be drawing over. Highlight the entire two layers above and press F6 to fill them with blank keyframes. These will be your drawing layers.



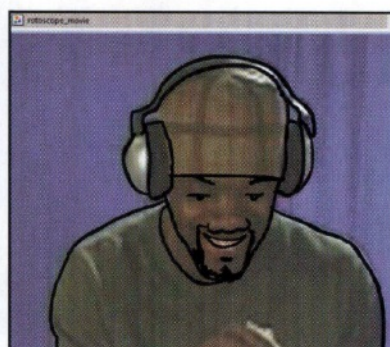
10 Now you're ready to start drawing. Select the Paintbrush tool. I select Use Pressure in the lower half of the toolbar to generate nice sweeping curves with a graphics tablet and pen. If you have a mouse, you can still use the Paintbrush on a small scale – it'll give you a different style, more like a ballpoint pen than a paintbrush.



11 Set the maximum width of your pen to medium, unless you're planning very thick curves. This is something you can experiment with to get the feeling you want.



12 On the layer above the video, select the first frame. Start tracing the outline of the image, ignoring the hands for now. Step through the movie with the , and . keys and complete all the frames. With features, simplification is a good idea – it saves time and, since the frames will be moving past quickly when animated, you won't notice if they aren't too detailed.



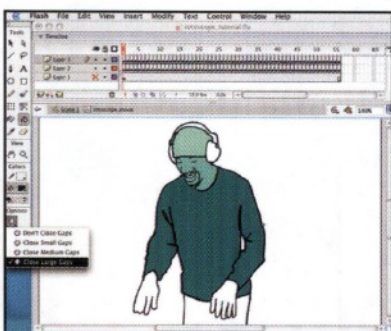
13 Concentrate on the key shapes in the face – the base of the nose, the eyelashes and iris (no need for the underside of the eyes), the top lip, mouth and the cleft of the chin. The overall body shape can be roughly sketched to add expression.

Speed drawing

To make the drawing process quicker and easier, it's often best to concentrate on one area at a time as you go through the movie. Your eye will grow accustomed to a particular shape and you'll acquire a feel for its movements. The , and . keys are useful for stepping quickly back and forth between frames.



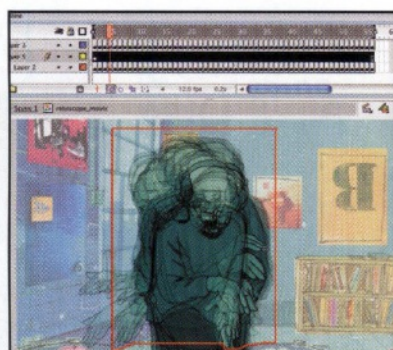
14 Draw the hands on the frames in the top layer. This will mean you can later make them interact with the scenery. Use the occasional line to hint at the fingernails, but don't worry about being too intricate. Try and make sure all your lines join up, so colouring in the figure later will be easy.



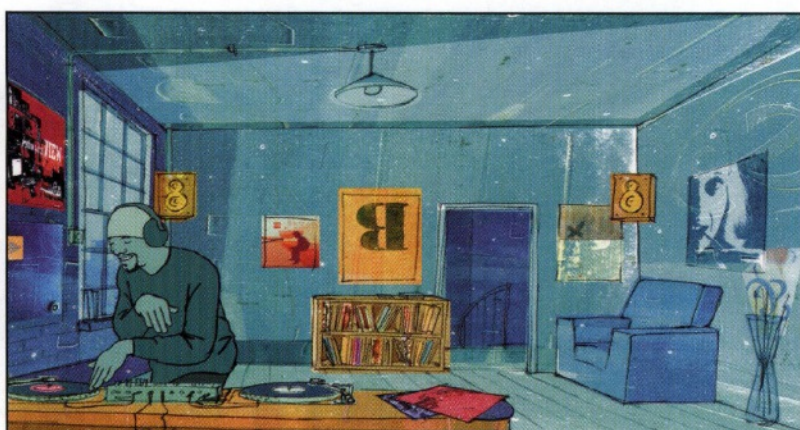
15 Once you've completed all frames, you can go through and fill them with the Paint Pot tool. If your lines haven't joined up in places, select Close Large Gaps from the Toolbar, or create a new layer underneath and block the colour in with the Paintbrush.



16 When you've finished, select the bottom video layer and delete it. Return to the main Scene, create a new layer underneath the character Graphic and import the Rotoscope_background.jpg. Match up the two elements by scaling your Graphic symbol. Make the length of both clips equal to the number of frames in your animation.



17 Now to make your character sit behind the foreground. Go back in to edit your Graphic and add a Mask layer above the main body layer. Leaving the foreground shape out, draw a mask over your figure that will contain all his movements (stretch Onion Skin across all frames to check). Mask the layer and your character will appear to sit behind the desk while his hands remain above it.



18 Now the animation is complete, you can import dialogue recorded during filming or dub in music, using Flash to edit. If your final file is for broadcast, it's possible to export it as a QuickTime movie for editing in another app. If you want to put it on the Web, Optimaze! (www.vecta3d.com) is a great piece of software for cutting down vector file size without sacrificing quality.

100% ESSENTIAL 0% DULL

THE WEB DESIGN BIBLE!

Discover how to create stunning web sites every month in **.net**

Inside each issue:

- ❖ Practical tutorials
- ❖ News and reviews
- ❖ Expert techniques
- ❖ Web building software
- ❖ Design inspiration



Find out more at www.netmag.co.uk

Make an icon font

If you're willing to push your imagination and design skills, it's possible to create a font every bit as personalised as fingerprints or DNA. Substance's newest recruit, **Chi-Yun Chen**, constructs a new font, G4alien, brick by brick

The concept behind G4alien was to create an iconic font based on my own childhood photographs, images with plenty of character and scope for development. When creating fonts, it's important to choose the right image, after all, this will be the image that propels your alphabet forwards and acts as a template or matrix.

This tutorial aims to underscore the importance of typography as a communication tool. When approaching a font, think of its application and relevance. Where will it be used? Continuity is key when creating a font or icon that will endure. Treat typography as you would a corporate logo. Iconography must be built from A through to Z and each letterform should maintain a unity with its predecessor while maintaining its own individuality. Most importantly, each letter must be both distinct and legible.

While constructing G4alien I wanted to fuse emotion with icons. Flash MX proved an excellent medium to realise that ambition, and to create a fonts matrix from my original childhood photograph.

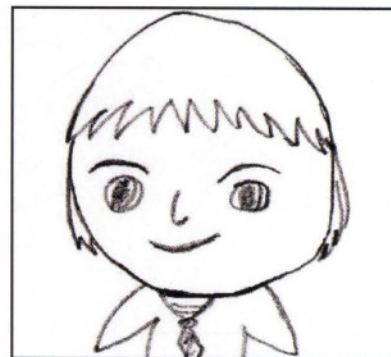
To maintain character consistency, a select number of line widths were chosen using only Line or Pencil tools to sketch each icon. The Flash MX Layers function provides underlying reference material at all times, a key benefit during character development. This function will also help you to maintain each icon's key components while giving you license to expand upon each icon's visual identity. So pay attention to the details needed to separate icon from icon or letterform from letterform, as it is these characteristics that will make your typeface truly unique. ■

Fonts: Making an icon font

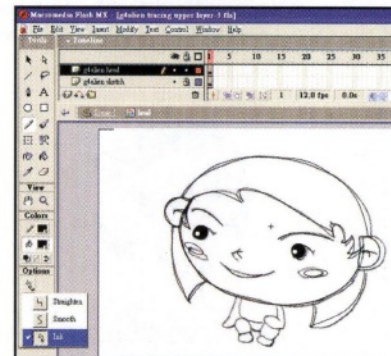
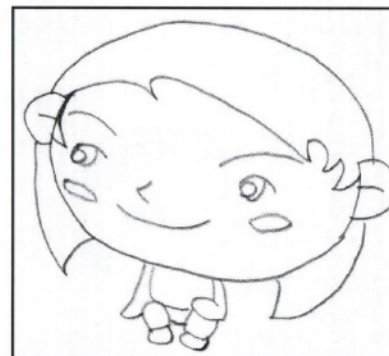
Constructing G4alien...



1 This original childhood photograph was the starting point for G4alien. The image has a wealth of emotion and cuteness and really stood out. When creating your own font it's important to select an image with lasting resonance. Print out several pictures as black and white laserprints at the same size as your source material.



2 Once you've selected an image, it is important to augment and hide some of the characteristics within your face. Embellish while sketching and focus-in on what makes your face distinctive. With G4alien we wanted to retain the innocent and vibrant look, characteristic of far Eastern pictography.



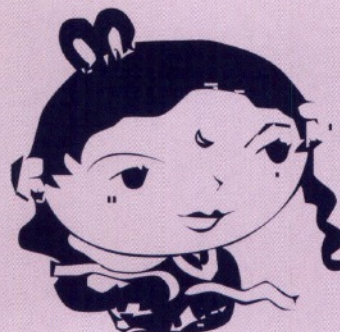
3 You can now define your original sketch. Using your first sketch as a background template, place a piece of tracing paper over the image and start to develop a sense of unity. Streamline and define the image using broad strokes. Scan in Photoshop and save the image as a greyscale jpeg at 300dpi with no compression.

4 Open Macromedia Flash, or your preferred drawing program, and import your scanned picture as a bottom layer (G4alien sketch). Create a new upper layer (G4alien head) and then use the Pencil tool to trace over the picture. Make any refinements using the Arrow tool.

Expertise provided by Chi-Yun Chen
You can contact Luke at
www.imagesofsubstance.com



恭

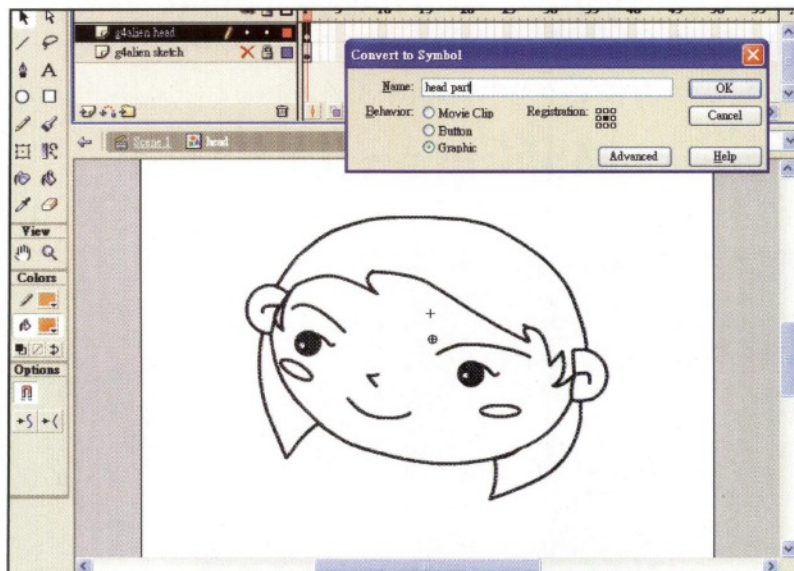


棋陳
昀

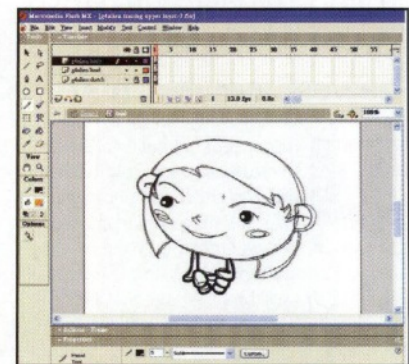
g4 外務人
ALien



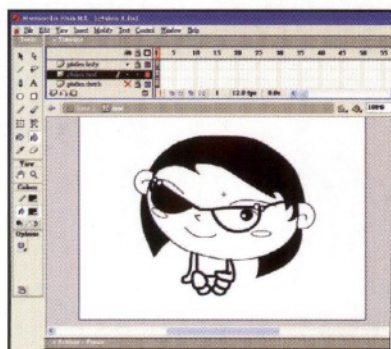
Fonts: Making an icon font continued



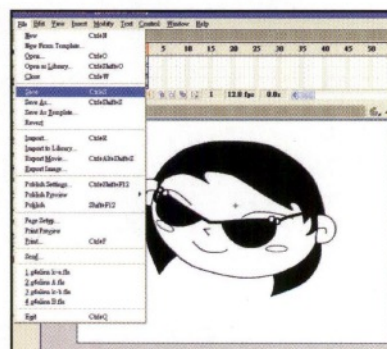
5 When using the Pencil tool select Options > Ink, this will make your image look less jagged. It's also worth locking your imported scanned picture (bottom layer) to prevent your image moving. Hide the bottom layer (G4alien sketch) and select the upper layer (G4alien head). Select Insert > ConvertToSymbol, name the file Head Part and select Graphic. This will give you your finished head icon.



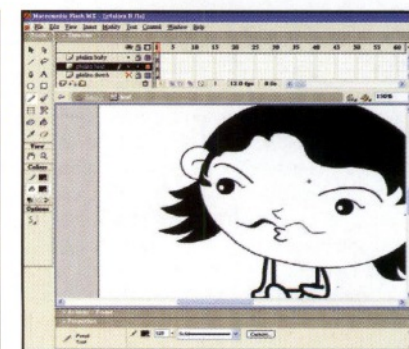
6 You can now incorporate a secondary design element. A head icon will represent lowercase and a head with a body will be uppercase. Insert a third layer (G4alien body) and label appropriately. To create the body, use the same procedure as steps 4 and 5. This will give you three distinctive layers – G4alien sketch, G4alien head and G4alien body.



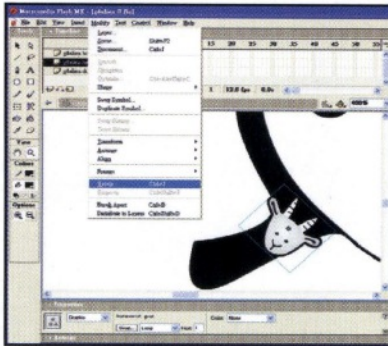
7 To create Cap A, first save your file by choosing File > Save as... (G4alien A). Lock Layer 1 (G4alien sketch) and Layer 3 (G4alien body) and hide your sketch layer (G4alien sketch). Concentrate on developing the head (G4alien head). In this example G4alien is wearing sunglasses. Remember to Save your file before continuing to develop your next character.



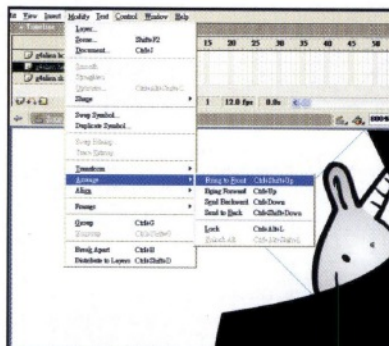
8 Next, create a lowercase A. Again, save your file first (G4alien lc-a). Before commencing work on Layer 3 (G4alien body), remember to unlock it. Then select the body layer with the Arrow tool and Delete. This will give you a complete head icon (G4alien lc-a). Save the file and return to the master template (G4alien tracing upper Layer-3) and continue to develop your alphabet.



9 Now work on uppercase B. Open the master template (G4alien tracing upper Layer-3) and Save as G4alien B. Adding additional features to each character will help you to develop a more attractive alphabet. Remember to Save your file at every stage. It's important that you have 26 lowercase files (heads) and 26 upper case files (head+body).



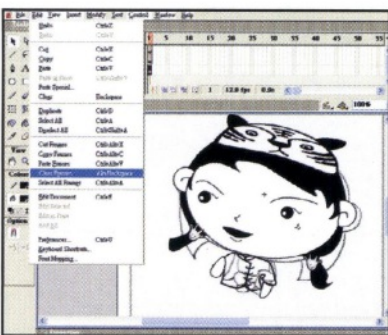
10 While constructing (G4alien G), we drew an interesting design detail using the Pencil and Oval tools. It is important to group elements. Go to **Modify > Group** so that individual elements can be arranged within the same layer.



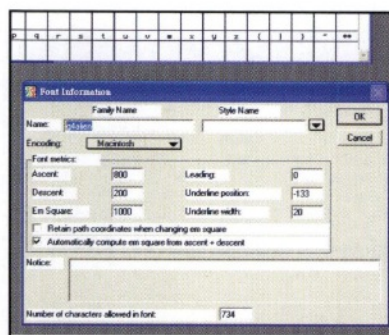
11 Arranging elements into grouped units can be of benefit when constructing your image. Select any of your grouped units and choose **Modify > Arrange > Bring To Front** to avoid overlapping, or **Send to Back** to establish an icon created with depth of field.



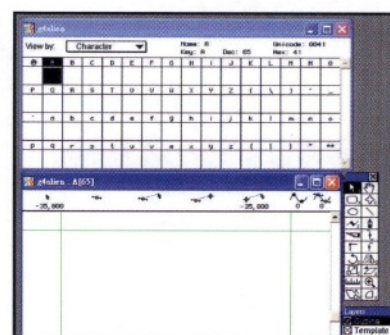
12 Before opening Fontographer 4.1, select Layer 1 (G4alien Sketch) and make it visible. Meanwhile, lock Layers 2 and 3 (G4alien head and G4alien body). You can now create a new file in Fontographer.



13 When deleting layers, it's important to remember to lock and save any layers containing vital information. Proceed by selecting your unwanted layer and delete it from the illustration. To delete G4alien sketch from the illustration, go to **Edit > Clear Frames** and Save the file.



14 Open a new file in Fontographer and name it after your font name by choosing **Element > FontInfo > General**. In the Font Information box, drop down the Encoding list to choose your operating system, either Mac or PC. Now save your new font file in a separate folder to your Flash characters.



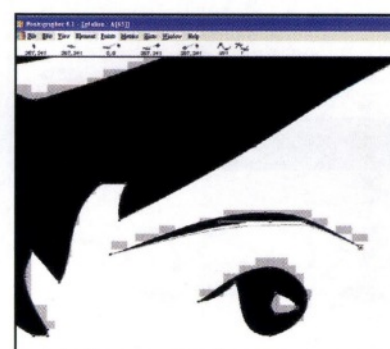
15 Once you've created a Fontographer file, select the uppercase A cell by clicking it in your Fontographer character window, which should be highlighted in black. Open your Flash file (G4alien A) and go to **Edit > Copy** to copy all elements. Go back to your Fontographer file and double-click the uppercase A cell.



16 Paste the character in a central position within the cell window and highlight **Template** in the Layers box. Go to **Edit > Select All** to frame your character and re-size the frame to fit within the guides. Repeat this throughout the alphabet to create 52 cells. You can add punctuation marks or numerals to your font at this stage if you wish.



17 Now select **Outline** in the Layers box. Go to **Menu** and select **Element > AutoTrace** where you'll see an option to create **Easy** or **Advanced** modes. The Easy mode includes everything that is usually required for an iconic font. Set **Curve Fit** in the Auto Trace box to ten. Select **View > Preview** and your filled character will be displayed.



18 You can now clean up the edges and any distortion left by your auto shape tracing. Next, select **Points** to help select an option to better suit the shape of your original image. For example, in **Points** select **Curve Point** to help soften edges or faces.

Optical spacing

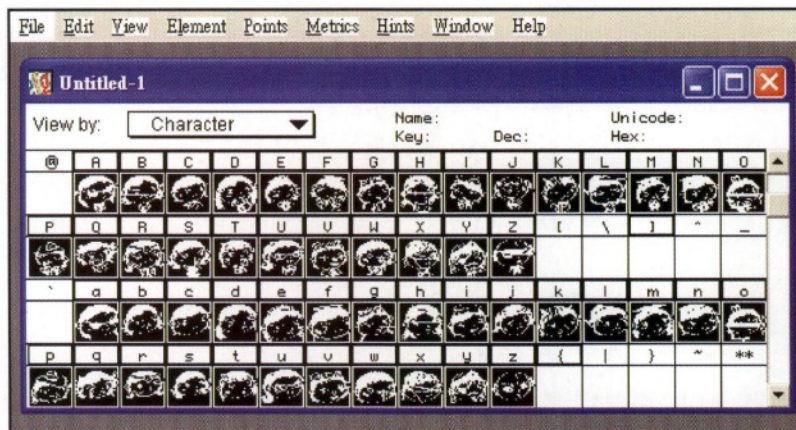
Pay attention to how your alphabet appears optically. The contrast between (black) icon and white space must be equalized as a composition that appears too tight will undoubtedly distract the reader and hinder the reading process. These judgments need to be made by human eye, as each character requires independent kerning. Experimentation will improve your typography — it's not possible to exhaust the potential combinations within any given font.



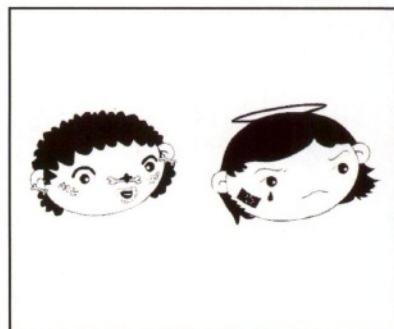
Fonts: Making an icon font continued

Arranging layers

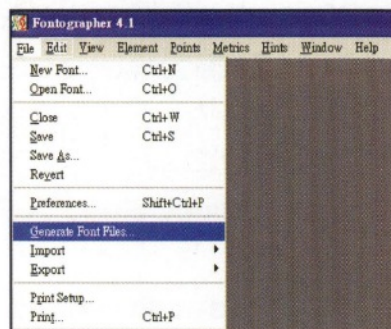
It is vital to create individual layers to create a sense of depth within your character. Layering your image will avoid any awkward overlapping caused by too many details on one layer. For G4alien we divided the icon into three individual layers: G4alien head (Layer 2) body (Layer 3) and sketch (Layer 1). Flash MX enables you to group elements within each layer. In the lowercase A featured here, G4alien is wearing sunglasses, which were drawn using the Pencil and Oval tools and grouped as a single image. Use Windows' quick keys to assign elements: Ctrl+Up brings your image forward and Ctrl+Down sends the grouped image backward. Should you then wish to edit a grouped image in isolation, you can use Windows' quick key Break Apart (Ctrl+B) and assign any specific component backwards or forwards.



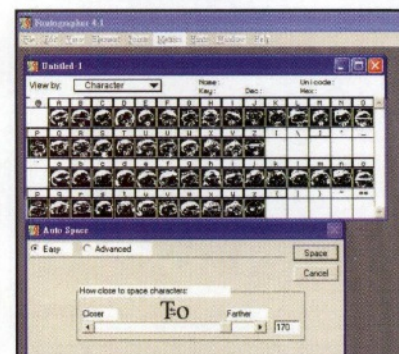
19 Once you have refined your entire alphabet, highlight all the character cells in your font case. Go to Menu and select Metrics > Auto Space to adjust the character spacing. There are two approaches, Easy and Advanced. Easy mode was applied to G4alien because of its iconic format.



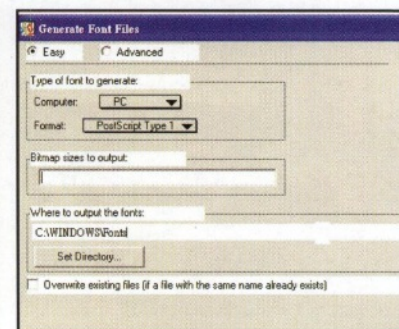
21 G4alien is a display font, so there's no need to develop punctuation marks or numbers. If you have spaced your font effectively, there should be enough white space around each individual character to let the image breathe. If the icons are too close, it looks messy and will have a disorientating optical effect.



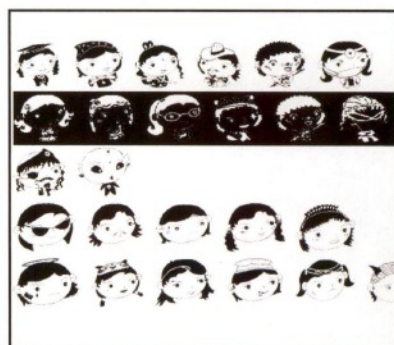
22 Your typeface will need to be exported as a PostScript or TrueType font. A TrueType font requires only one outline file, while PostScript needs an outline file and a secondary file. Before starting to incorporate this into the font folder located in your system, remember to Save and then go menu to File > Generate Font Files.



20 Since the G4alien font has a wide character width, set Auto Space to 170. A more traditional font will need a closer fitting. In the Metrics table there are more individual options to help you kern effectively. You may need to repeat the process of kerning individual characters several times, so keep checking the space between your characters.



23 Name your font for screen application. Place the font into your system font folder. Select either Mac or PC type and choose your preferred Format to output – PostScript Type 1, PostScript Type 3, TrueType, or none. The Set Directory will help you search, select an Output Directory and Generate the font within your system font folder.



24 Once you have successfully imported the font into your system font folder, check the spacing and character definition by displaying it in any drawing program. Does your font have a sense of unity? If not, refine the character spacing in Fontographer's Metrics table and regenerate following step 23.



25 Once you are happy with your PostScript or TrueType font, you can start to use it creatively. In this example, we have used G4alien G as a basis for a composition and coloured it in Flash to create a composition unique to the character.

Pe motion

Clusta Design Consultancy shows how type can be used to add meaning to a short Flash video...

The following tutorial explains how we went about combining text, imagery, graphics and video footage to produce an intro to the magazine's 'creativity', and we start with

Fhe following tutorial explains how we went about combining text, imagery, graphics and video footage to produce this short for Create-Online magazine. The video was used as the intro to the magazine's covermounted CD-ROM.

Fhe following went about creating graphics and video in this short for Create-Once-used as the intro to the mag covermounted CD-ROM.

The theme of the video was 'usability' and we wanted to communicate this idea with fast and furious graphics, using words that we come into contact with every day while using computers. The video employs techniques that we use in almost all of our video production work. As a web design company, using Flash for video work seemed like an obvious solution, and this tutorial demonstrates the versatility of Flash as a tool for producing video - 80 per cent of this piece was produced in Flash and then exported to After Effects to piece together with the music.

Typography is an important part of all our work, especially video production. This tutorial explains why we've used typography in this way and how we went about animating text for video production.

Expertise provided by Clusta Design Consultants
You can see more of its work at www.clusta.co.uk
To contact the company at hello@clusta.co.uk

Expertise provided by Clusta Design Consultancy.
You can see more of its work at www.clusta.com
and contact the company at hello@clusta.com

The finished Flash movie feature can be found on the cover disc.

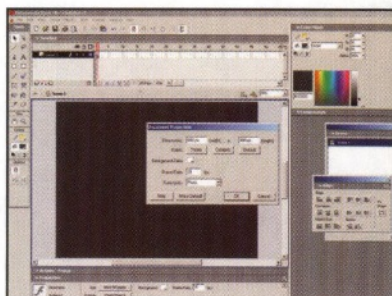


Part 1: Kicking it off

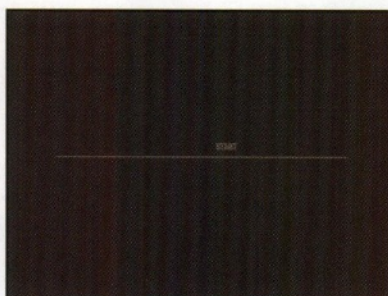
First up, you need to prepare the document and animate the typography...

Exporting frames

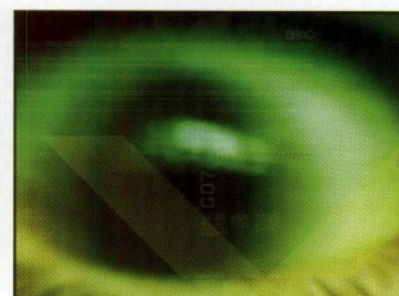
It's vital to note that when exporting frames from Flash, embedded movie clips with their own animation timelines will not be recognised as animations – they will simply be exported as the first frame of its animation. Keep all your animation on the main timeline, so that when you export, you'll see everything you've done.



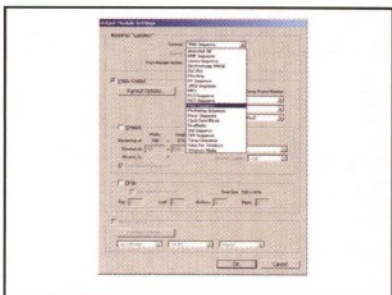
1 The first thing we do is open Flash and set the movie dimensions. Because this is for a CD-ROM, we select 640x480. The background is created in Flash by making a grey box that sits on the timeline underneath all other layers. We type the word **START** (using Impact) into a text box on the timeline.



2 To create the impression that the **START** text is moving quickly, we create a gradient-filled object in Flash the same size and colour as the word. We motion-tween this gradient from the top of the screen to the point where **START** appears, and replace the gradient with **START**. In this instance, we want the word to be fairly small, just about legible, to give more impact to the main visuals once they kick off.



3 For the next section, we film some close-ups of eyes, which are taken into After Effects and adjusted using colour balance layers. Two bits of footage are put back-to-back here – by motion-blurring the first bit of footage at its end point, it flows into the second piece smoothly.



4 This footage needs to be exported as individual frames, so we go to Make Movie in After Effects and click on Loseless. This pulls up the Output Module settings dialogue. At the top is a drop-down called Format. From here, we select PNG Sequence. Now we render our individual frames to a folder.



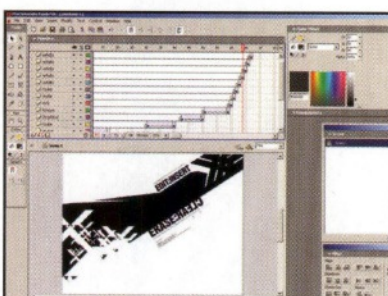
5 The frames are imported into Flash by using File > Import. We select the first frame and click Open. When asked if we'd like to import the whole sequence, we say 'yes'. The footage is then dropped on to Flash's main timeline. The first frame of the footage is made into a Flash movie clip and is motion-tweened using a colour tint transition from white to its own colour.



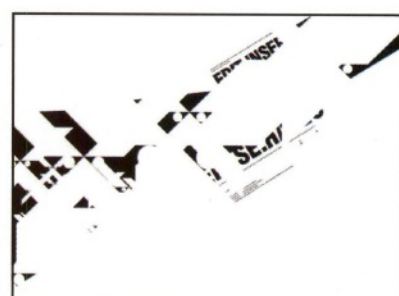
6 At this point, we want to convey the idea that the eye is being consumed by graphics/imagery. We grab some old vector files, record decks, bits of fonts, and so on, and layer them on top of each other until they almost cover the eye. These are then divided into five layers and animated into the scene in chunks.



7 We want to add a piece of graphic at this point. The process for this is to create the graphic and work backwards, animating it into the scene. We use Impact as the font choice because it's bold and doesn't get lost against the stark black graphic overlays.



8 The elements of the graphic are broken down into parts and then motion-tweened in Flash, either as colour fades or along motion paths.



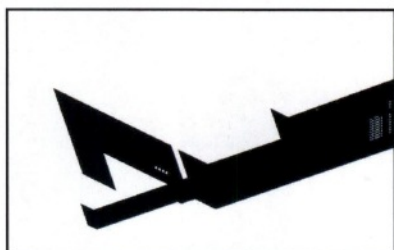
9 We now want to move out of this scene, so we need a transition. The logo, designed by Mezzetty, is already part of the graphic and seems an ideal symbol to overlay in white repeatedly until the graphic had disappeared. This enables us to move on to our next scene fluidly.

Part 2: 3D drop-downs

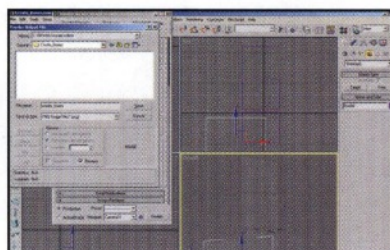
Produce 3D elements in 3ds max and incorporate them into the scene...

Understated effects

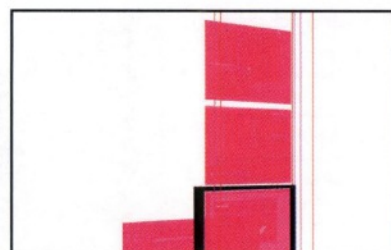
Try not to be tempted to use the traditional Effects options in Photoshop or After Effects to enhance your footage. Often the best-looking footage uses simple blurs and colour tints. By adding your graphics or typography over the top with some animation, you can add all the energy and style you want without using obvious filters.



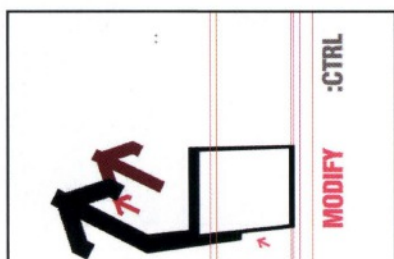
1 This scene again focuses on a central graphic and then builds around and adds to it to give it energy. We start with the core component of the graphic in black and then introduce a 3D element.



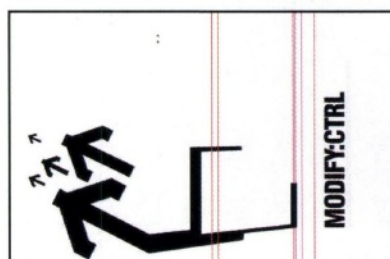
2 The 3D element is produced in 3ds max. It's a simple set of boxes animating into the scene using ease curves. Each box is mapped with a screen grab, and a small amount of Opacity is used to add extra depth. The animation is exported as individual .PNG files, which gives you an alpha channel and more options to manipulate each frame once in Flash.



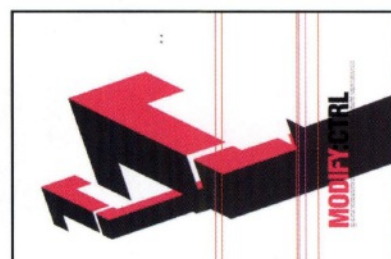
3 The single PNGs are imported into their own layer on Flash's main timeline. It's the second layer up; the bottom layer being our core graphic. To integrate the 3D with the 2D graphics being created in Flash, we design an arrow graphic around one of the boxes in Flash and then duplicate this frame 30 or so times. We then work backwards deleting parts of the graphic until it looks like it's animating itself in.



4 The words Modify:Ctrl are added to the scene using the same font to retain consistency; they are introduced down the right spine to coincide with this section's flow.



5 The arrows are animated out in the same way they're animated in, and we introduce some lines to the foreground to add interest to the graphic. These are drawn in Flash and motion-tweened into the scene.



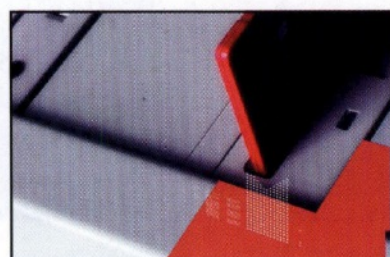
6 We now re-introduce the core graphic but in its full form. The graphic is separated into several elements and divided between layers, which then enables us to motion-tween colour tints randomly across the timeline until they fade to white and we can move to the next sequence.

Part 3: Floppy animation

Film a disk ejecting from a drive and animate the video footage in Flash...



1 For this scene, we film a close-up of a diskette ejecting from a drive. Some of the frames are taken into Photoshop and motion blur is added to them at random positions.



2 In this instance, the graphic is built around the video clip, and is triggered by the diskette ejecting. The graphic is built up and separated into layers for animating. We focus on the word 'User' and add very small text to accompany the elements. This gives the graphic a tech feel, like programming code.



3 The floppy drive is divided into its more interesting shapes by creating new layers in Flash and tracing the various elements. Each shape is filled and turned into a movie clip. The movie clips are then animated into the scene either as colour tints or motion-tweened along paths.



Part 4: Bringing in more human elements

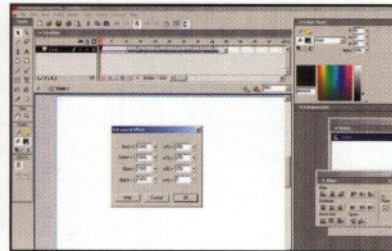
Now you need to work with the still images to create movement and motion...

Photoshop actions

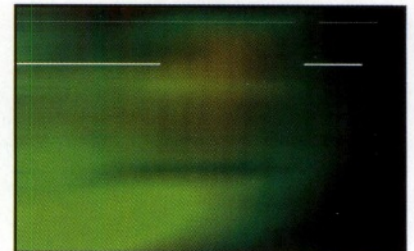
When you're dealing with small chunks of frames like we have in this tutorial, actions in Photoshop can be an incredibly useful tool for colour-tinting or blurring frames. Simply open the Actions dialogue box and create a New Action. On the tab where it says Function Key, select F2 from the drop-down. Now record your actions by applying tints, and so on, to your image. When you've finished, press the stop button next to the record button. If you open up another frame and press F2, you'll see it applies the action you have recorded.



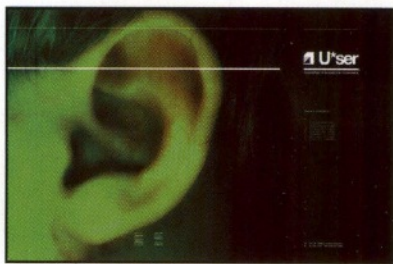
1 This scene only requires a still shot. We introduce some colour-tinting in Photoshop to give it a green, old-skool VDU sort of feel.



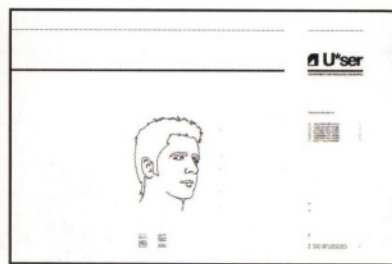
2 We bring this in from white, which gives it more impact. To do this, make a movie clip from the first frame of the animation and do a colour-tint motion tween, and use the 'advanced' tint option and set the RGB values to full. When they normalise through the tween, you get a bleaching effect.



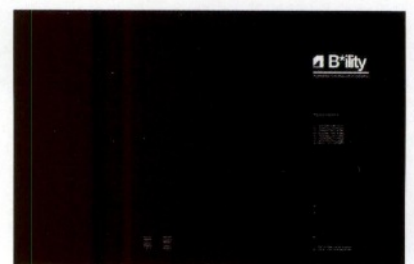
3 By motion-blurring the frames in Photoshop before bringing them into Flash, you get the impression of movement. While the ear is blurring and moving to the left we introduce a couple of graphic elements that are motion-tweened in Flash from the left-hand side to emphasise the motion.



4 We build up a graphic piece from this still because it gives us plenty of space to build on the look and feel of this shot. The word 'User' is now the solid theme for this piece and we want to develop that idea further by creating an almost brand-like feel for it. So we create a logo from it.



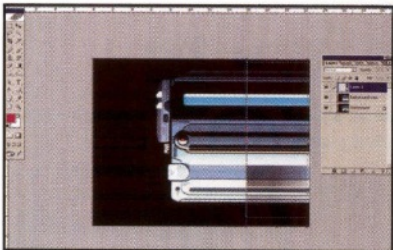
5 The ear is bleached out in Flash and we're left with the graphic components. We add in an outlined image to the graphic to form a tie between the ear and the 'User'. The structure of the type is laid out in such a way as to give a machine-like quality.



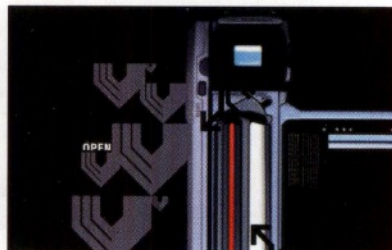
6 Still in Flash, we flip all the colours using colour tints and change the word 'User' into 'Bility' as a continuation of the overall theme. This then gives us the opportunity to drop out the text frame-by-frame so we can move to our next section.

Part 5: Digital extrusions

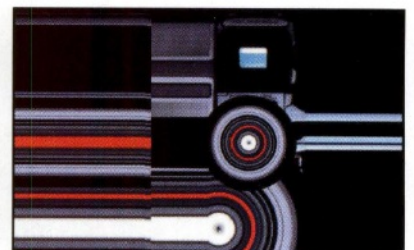
Stretch the stills to create abstract and interesting footage...



1 This section is again made from just a still image. The main technique for creating movement in this scene is produced using Photoshop. We take a still of a video camera and open it in Photoshop. We then select a very thin slice of the image, and copy and paste this into a new layer. Using the Transform tool, we scale this thin slice, which stretches the image from that particular point.



2 Once the frames have been manipulated in Photoshop, we can take them into Flash and overlay graphics and text. The start of this particular section is treated frame by frame, each frame being treated as a new graphic piece. Symbols and text are layered here in a way that works well with the manipulated camera shot.

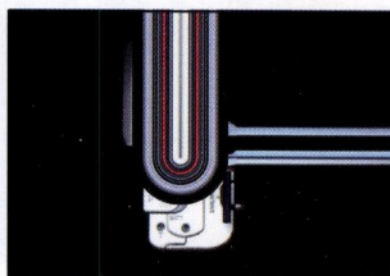


3 Here we use exactly the same techniques as described in Part 1 of this tutorial, but we introduce a series of rings to add colour and abstraction to the scene.

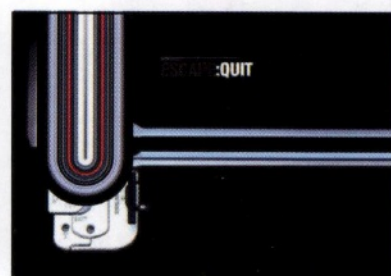
Digital extrusions continued...



4 Now we want to add some structure to the scene so we can introduce more text. We start by taking a simple abstraction of the camera from Photoshop and importing this into Flash. We create the 'extruded rings' section of the image in Photoshop as a separate element and save it out as a PNG. This is imported into Flash on a layer above the camera image.



5 By masking the 'extruded rings' layer we can animate the rings growing out from the camera by using a simple motion tween. Because we're only doing this at 72dpi, we can take a screen grab from Flash and take it back into Photoshop. From here, we can motion-blur and shift its position to give the effect of motion.



6 Once in position and back in Flash, we can start adding some text to the graphic. We're continuing with the same theme for the words and styling, bold headline text with small text elements to give it a techy edge. We animate the text with simple colour tints to highlight the words as they appear.

Part 6: Keyboard mapping

The final stage is to pinpoint moments to trigger graphic elements...

Tying in music

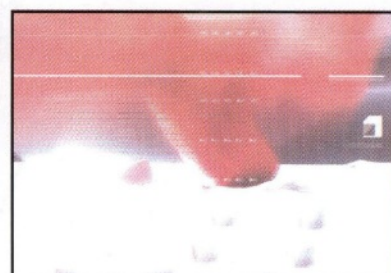
For most projects, once we've exported all our frames from Flash and imported them into After Effects, we'll key in the music by cutting up and delaying bits of footage until they're in time with the basic 4:4 structure. However, we've found that if the footage is quick enough, with constant change, you don't need to key in the music because it will work with the imagery regardless. Try it on a short bit of footage and music.



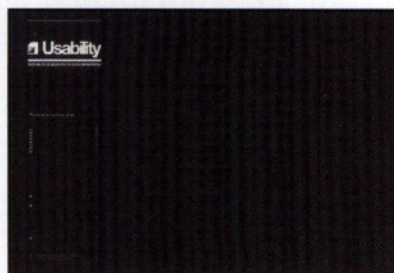
1 The footage here is taken into Photoshop and adjusted by making a duplicate layer, adding 5-per-cent Gaussian blur and overlaying it on its original. The footage is then imported into Flash. Graphic elements that we've been using throughout the piece are introduced to add consistency and give the footage a more interesting aesthetic.



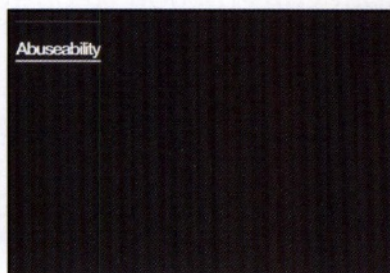
2 At the point at which the hand makes a keystroke we intervene with some graphics, which exaggerates the movement. This is simply three frames of differing shapes/graphics – there is no tweening here.



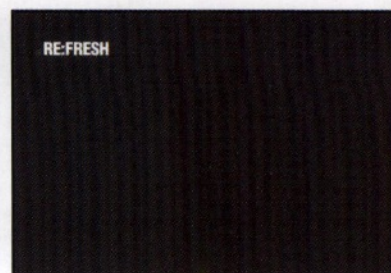
3 One of the individual frames of the footage is converted into a movie clip in Flash and we use the 'advanced' tinting technique to bleach it out. We also add some repeating number sequences to give the impression of a slight glitch.



4 We create a gradient in Flash and animate it to give the impression that the text is moving about quickly. We then animate in the rest of our elements – the symbol is scaled in, the two lines are animated in from the left and the rest of the elements appear over a staggered series of layers.



5 The word 'Usability' is replaced with 'Abuseability', as in to push your own abilities as far as you can.



6 Finally, this is replaced with 'Refresh' and we introduce the Clusta Gundamn and logo for final sign-off. ■



Flash MX: Typographic animation



All the components needed to complete this tutorial can be found on the cover disc.

From basic font construction to button states, animation and alignment with an XML-driven navigation structure, we take a look at the backend architecture behind the impressive planetofthedrums.com

The Planet of the Drums tour brings some of the loudest, most intense electronic music to over 25 North American cities. Coupled with imagery of damaged/destroyed audio devices, it is important that the web site's typography and navigation brings the same loudness to engage users in the experience. The solution? FontLab's side-program, FontFlasher, and the development of a fully dynamic and customisable navigation system (with tricks) in Flash.

This tutorial will show you how, using typographic animations in Flash, we came up with the effects you see on the web site.

The navigation works in an iPod-esque fashion, scrolling to the right to take you further into the site, and to the left to bring you back. To maintain the distortion, we gave the motion a random quality, moving at different increments at a time.

In order to better distinguish the white type from the underlying white image, we tried to almost separate the two with

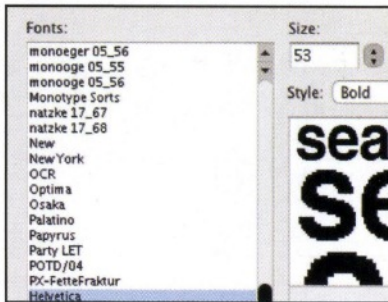
resolutions. As a result, the 85pt Helvetica Bold looks as though it could be 8-bit.

The navigation buttons were also designed to have various states, which included the up (standard white), inactive (hashed white, for forthcoming sections), mouse over (solid green) and visited (hashed green) states.

We'll show you exactly how these elements were produced both in FontFlasher and Flash, and leave you with the ability to use a very powerful resource for your future web site development. ■

Part 1: Font building

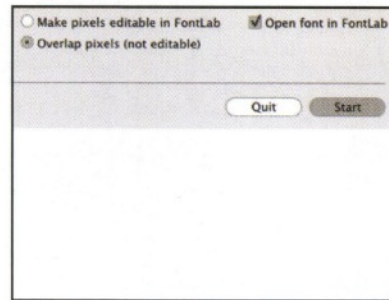
How to build fonts in FontFlasher...



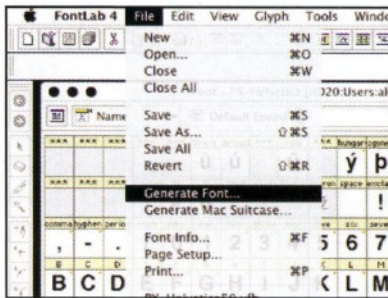
1 In FontFlasher, select an installed typeface and the point size you would like to Rasterize. The right-hand preview pane will now show you an aliased preview of your selection.



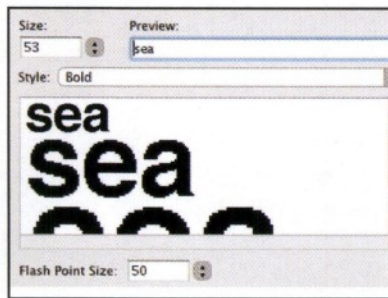
2 Select the Flash Point size. This is the root formation of the pixels and the point size you will need in Flash for your font to remain as crisp as possible. It's best to keep this point size similar to the original point size in the event of a font substitution.



3 By overlapping pixels you will make your typeface less complex and maintain a smaller-size font, though you should expect a much longer rendering process (especially when drawing larger point sizes). Check off Open Font in FontLab so your resource file will open immediately when finished.



4 Upon export, FontLab will launch and immediately prompt you to save the FontLab (.vfb) file. From here, go to File > GenerateFont. It's easiest to create a TrueType font, as this will work on both Mac and PC.



5 Install the font into your system and move it into Flash. For the chunky/8-bit effect, simply scale up your new font to a larger point size (preferably in multiples of the Flash Point size, to maintain crispness).

Font flasher

FontFlasher is a utility/side application to FontLab that converts almost any format font into a "pixel" TrueType font (which turns on only selected pixels). This gives fine text a much clearer, more legible onscreen appearance. For more details visit <http://fontlab.com>.

Flash development

The Flash navigation system consists of three main components...

NAV

This is the main navigation structure. It loads and stores all the XML data and creates sub-menus as necessary. It slides left and right to display the currently selected menu.

MENU

The menu structure populates itself with buttons from the Library according to the XML data it receives from the NAV. This Movie Clip is then duplicated to create sub-menus.

BUTTON

Each button contains a text field to display its title, a "textMask" Movie Clip to create a mask in the shape of the title, and an image called Screen that

is used to create the cross-hatched effect. The button has an Off, Over, and Clicked state, which is controlled by actionScript.

This system is mainly code based. The button is the only item that needs to be designed.

To give this information some context visit www.planetofthedrums.com, or take a look at the POTD_tutorial.swf file on the cover disc before you start.

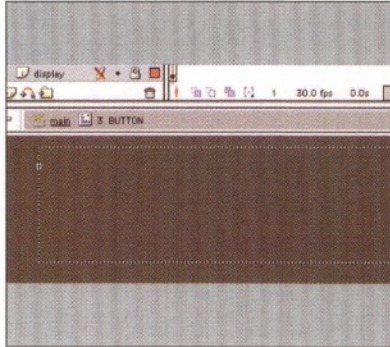


Part 2: Building the button

How to create your button...

The right choice

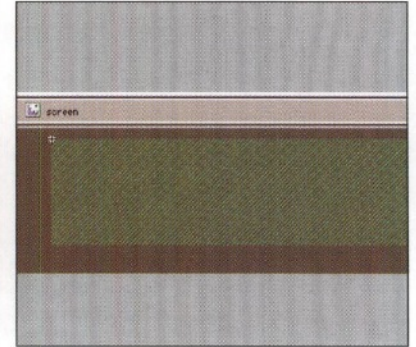
When deciding on the point size at which you'd like to Rasterize your font, it's usually best to have a pixel formation that closely represents the real font. We use the characters A, E and S as a gauge, since they are usually the more distinctive letterforms.



1 Create a new Symbol with Behaviour set to Movie Clip. Create a text field on the first frame of this Movie Clip. Name this text field "display" and then set your parameters (embedded font, html, selectable etc).



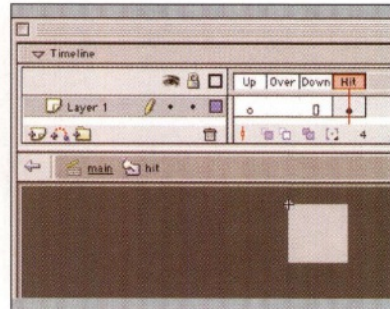
2 Copy that text field and place it on a layer below the original. Make sure that it lines up exactly. Select the new field, and convert it to a Movie Clip. Name this instance of the Movie Clip "textMask".



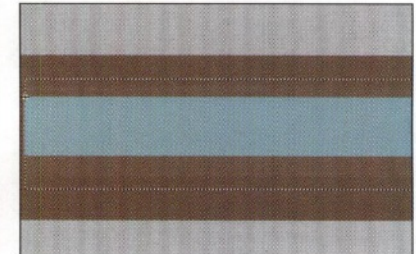
3 Import or create a graphic for your background texture. In the case of the POTD site, we used a .png image of diagonal pixel lines – transparent so that the background of the site is visible through the texture. Convert your graphic to a Movie Clip and name the instance "screen".



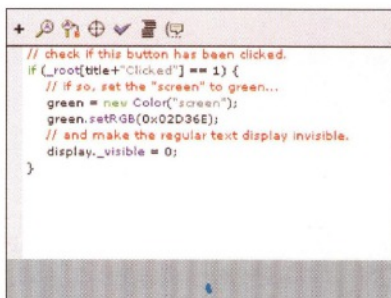
4 Place the "screen" Movie Clip on a layer under "textMask" and set the "textMask" layer to mask the "screen" layer. These two Movie Clips will create the Clicked button. The "textMask" symbol will not be visible, but the text field within it will contain the same text that appears in the Display text field. This will give you a mask shape that matches the Display text. When the button is clicked, the Display text field will be made invisible so that only a text-shaped mask of the background texture remains.



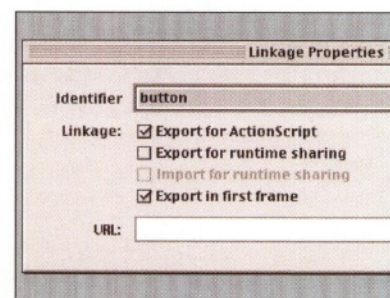
5 Create an invisible button that contains only a square on its "hit" frame.



6 Place this on a layer above the text mask. Adjust its size so that it covers the area you want to be clickable. Opt for an actual button rather than an onPress function to keep the hit area localised to the x-height of the font rather than the bounds of the Movie Clip. Add the mouseOver/mouseOut actions (found in the .fla file) to this button.



7 This button is designed to "know" whether it has been previously clicked. When a user clicks on a button, a unique variable based on the Display text is set at the _root level of the Movie. When a button appears on the stage, its first action is to check whether it has been clicked before. If so, it sets itself to its Clicked state.



8 In the Library, right-click on the Button symbol and select Linkage. Give it the identifier name "button" and set it to Export for ActionScript. Now this symbol can be drawn from the Library on demand.



Part 3: The navigation structure

This is key to the usability of your web site...



1 Load the XML data "content.xml". An XML object will be created and the information passed to the first menu, "menu0", which is already in place. Based on this information, the first menu will now populate itself with buttons from the Library (one for each node on the XML tree) and name them accordingly.



2 When a button is clicked, it will check to see if its XML node has any childNodes. If so, it will create a sub-menu (by duplicating "menu0") and pass its childNode data to the sub-menu. The sub-menu will be placed 510px (the maximum width for this layout) to the right of the main menu.



3 This entire Movie Clip will now slide horizontally to the left, so that the sub-menu is visible.

Still not crisp?

If you find that your font isn't exporting as crisply as it should be, set your Flash document to a lower quality setting. But be careful as this may jeopardise any curved vector illustrations in your file.



4 When the Back button (on the main stage) is pressed, this Movie Clip will slide to the right so that the main menu is visible. Any sub-menus are then removed. You can make your XML structure as deep as you like – the menus will duplicate infinitely.

The navigation/motion/animation

To match the lo-tech aesthetic of the site, we wanted the navigation to move in a rough, erratic fashion. Rather than sliding across the screen smoothly, the navigation shuffles across in random increments.

To do this, give the navigation a destination position, pick a number between one and 100,

and then move the navigation towards its destination by that number of pixels.

That covers the visual and conceptual side of the navigation system. For all the code, and a more detailed overview of the functions, check out POTD_tutorial fla, which you'll find on the cover disc.





Better characters

The Pocket Laboratories team, the brains behind the brilliant morejamsandwiches.com, explains how best to adapt your Flash knowledge to master character animation, and draw attention to your work on the web

Inspiration

If you need some inspiration when designing your new character, take a look at other web characters, comic book characters and animated characters on TV. Concentrate on what makes them likeable and appealing. Is it the body shape, colour or facial details? www.pictoplasma.com is a priceless resource. The company publishes books based on fresh character design and also holds international conferences on the character animation.

There's a huge emphasis on interactivity across the internet these days. Eye-catching, distinctive, interactive characters can be a valuable addition to any web site. So if you are looking for an entertaining interruption to your day-job, some odd little web characters might just be the perfect distraction. Whether they exist in an interactive animation or an addictive little game, first-rate character-based web sites gain cult and mainstream followings on the net and are a great way to draw attention to your web design work.

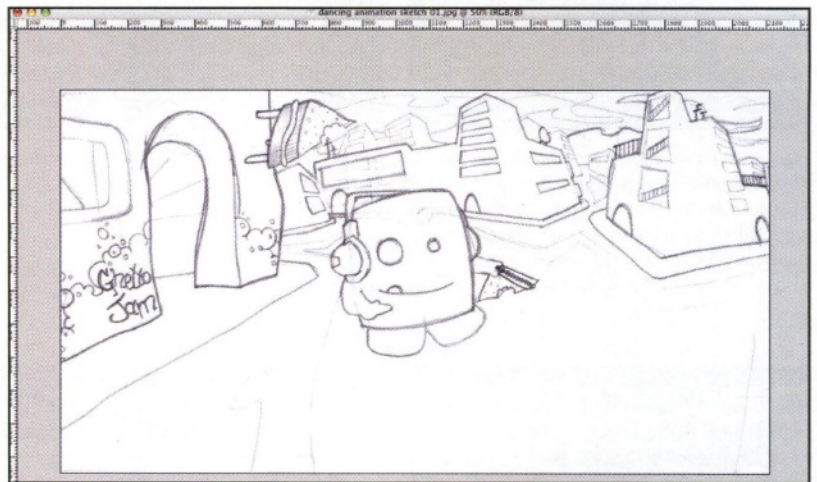
The secret to creating a successful interactive web character lies in the balance between fun and reality. No matter how insane or unreal your character becomes, you will reach a greater level of popularity if you add perfect mannerisms to bring that character to life, to build on the relationship between your character and the user.

Interactive character-based sites can entertain and educate, there's an open opportunity to charm children and adults alike. Characters can be produced in an endless array of styles and, if created skilfully, add a magic element to a web site, be it corporate, conventional or contemporary.

This tutorial will teach you how to adapt your Flash knowledge to master character animation. Step-by-step you will learn how to take a character idea through the process of design, artwork, animation and interactivity, essentially bringing your character to life. The final product will be a fresh and interesting animated character with which the user can interact fully and directly. ■

Animation: Characters

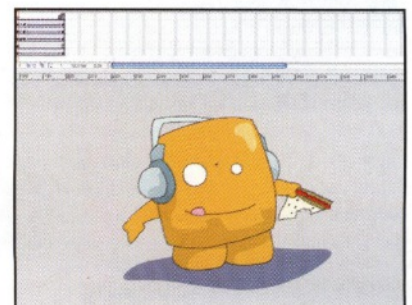
Start with a pen and some paper...



1 Start by developing your character using some good old pen and paper. This may seem old fashioned, but it is important that your character is likeable and this preparation will help to give your character the originality it needs to stand out from the rest.



2 There are some important basics to bear in mind when designing your character. It'll need an appealing body-shape, clear and simple arms and legs, and a characteristic yet simple face (whether it's cute, dark or outrageous). As soon as you start to animate the face and limbs you will be thankful they were kept simple.



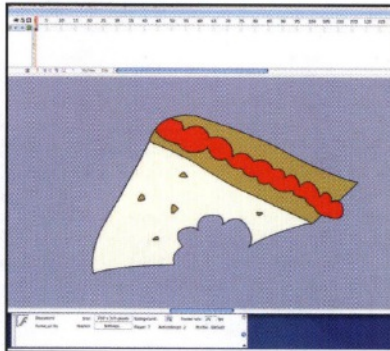
3 When developing your character in Flash, ensure you keep each element on a separate layer right from the start. This will make the animation process much easier. Generally, you should use a new layer for the body, each leg, each arm, the head and then different layers for the eyes and mouth, too.



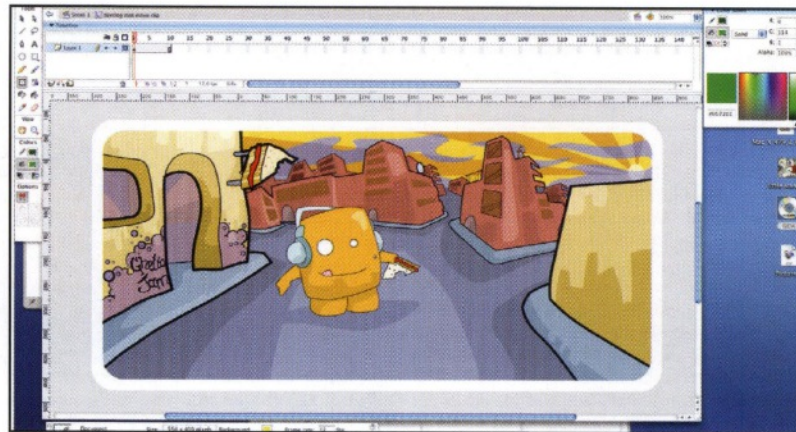
Expertise provided by Pocket Laboratories
You can contact Pocket Laboratories at
www.pocketlaboratories.com



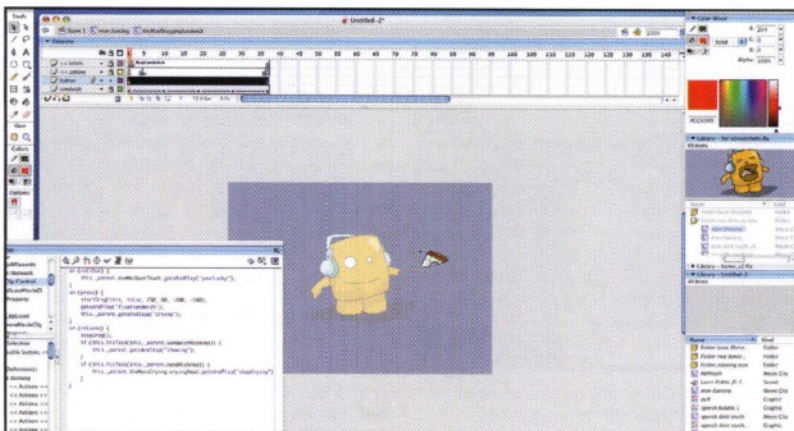
Files on disc
You'll find the files you need on the cover disc.



4 Uncomplicated and understandable interactive props are always effective. For this tutorial the prop will be a jam sandwich. The user will be able to interact with the character and jam sandwich individually and then make them interact with each other.



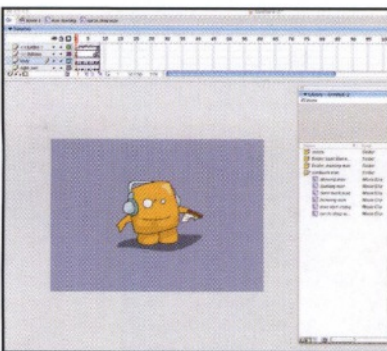
5 A simple background can add so much to your scene and create a sense of space and depth within your web creation. When colouring the background, props and character bright colours are generally best, but it pays to steer slightly away from the primary colour palette for a fresher appearance.



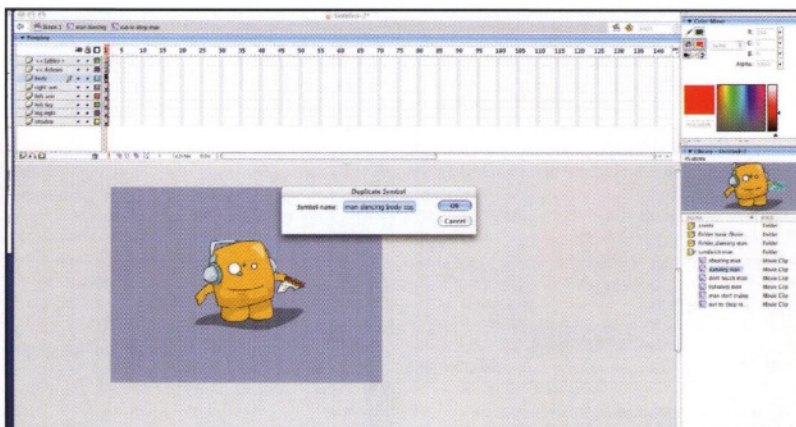
6 There are several movie clips of the sandwich, each with a different set of actions depending on what the character is doing. For example, the sandwich the character holds while dancing is a different movie clip to the sandwich the user drags around the screen while they interact with the web site.



7 Turn all the character elements (legs, arms, eyes, head and tongue) into graphic symbols. This will allow you to animate each element separately. Drag-select the entire character, turn it into a movie clip, and name it "Dancing Man".



8 Duplicate the Dancing Man movie clip five times and name the clips Don't Touch Man, Crying Man, Chewing Man, Run to Shop Man and Listening Man. These movie clips will each be made into separate animated sequences. Place an actions layer and a labels layer at the top of each movie clip.



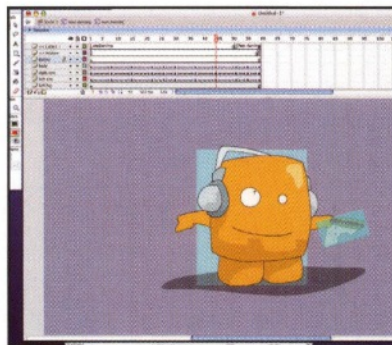
9 Within each movie clip the animation will take place within the head while the arms and legs stay still. Go into each of the six movie clips and duplicate the head, renaming it accordingly (Dancing Head, Don't Touch Head, Chewing Head etc).

Before you start

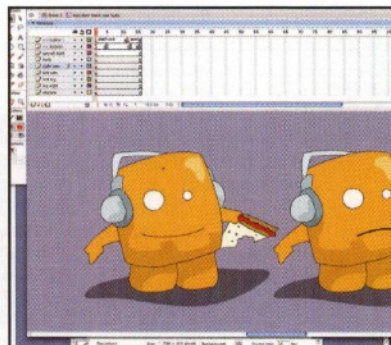
Plan how your character is going to move before starting your animation and be sure how it will interact before you start scripting. Keep your library tidy while you work by placing all of the symbols you create into folders with relevant names. This will save time when you try to find them again later.



Animation: Characters continued



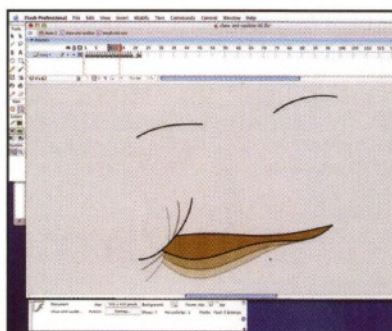
10 Dancing Man. This character will need to loop, so it is important that the first and last frames are exactly the same. Think through how it should move, taking into account its size and shape. Now Motion Tween the separate body parts in order to obtain the right dancing movement.



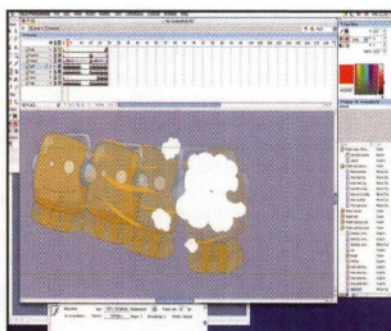
11 Don't Touch Man. Now animate a sequence where the character changes from a happy expression to an unhappy expression and back again. Try playing around with different facial expressions, mouth and eye shapes as well as the character's general posture.



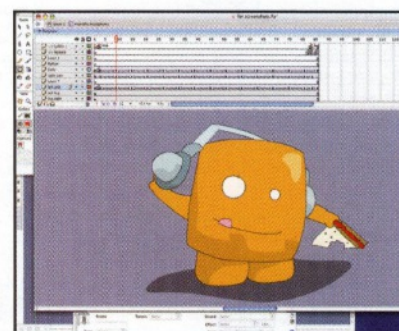
12 Crying Man. To achieve a more natural reaction, animate a transition sequence to go from unhappy to crying, to looking unhappy again. The crying sequence should be a three to five frame loop that will repeat continuously until another action takes place.



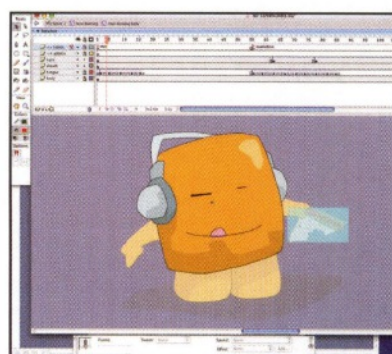
13 Chewing Man. Use frame-by-frame animation to create a more realistic movement than you could achieve with tweening alone. Use the Onion-skinning tool to double check you have created the correct movement from one frame to the next.



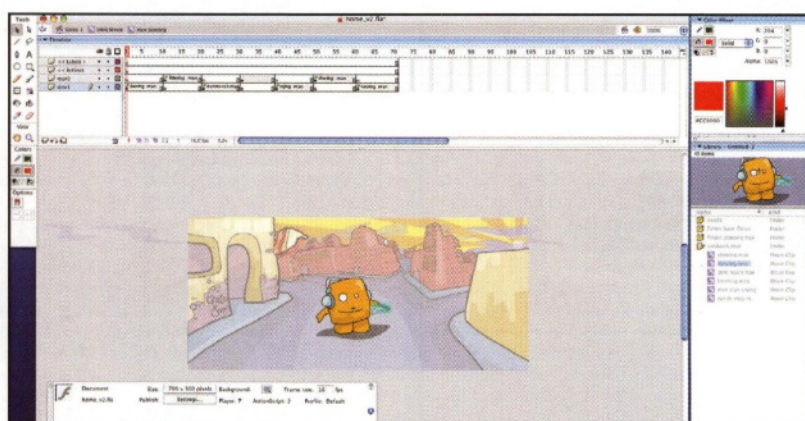
14 Run to Shop Man. Now that he has eaten his sandwich your character will need another one. In the "run to shop" sequence use clouds of dust, movement lines, and alpha transparencies of the character to create the illusion of fast movement.



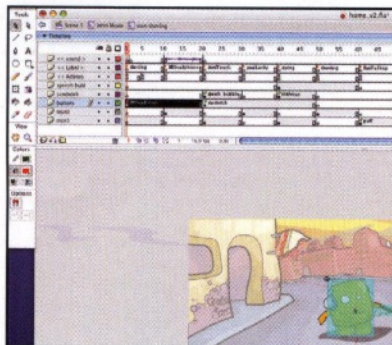
15 Listening Man. In this movie clip the user interacts with the character directly rather than involving the sandwich. Create a short transition sequence where he lifts his headphones followed by an animated loop of him dancing. Finish off by making him put his headphones back on.



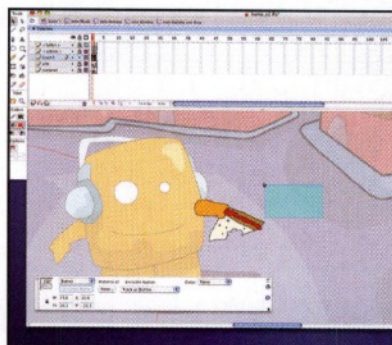
16 Add small characteristic touches such as blinking and lip licking to bring your sandwich man to life. Placing animated elements within the scenery will also add interest. Try animating the clouds, switching street lights on and off or maybe even adding background characters.



17 Create a new movie clip named "sandwichMan". Drag all six movie clips from the library onto the timeline in order and make sure that each movie clip is ten frames long. You will need to align all of the characters so that they are placed in exactly the same position on the stage.



18 Select the labels layer and place a label at the first frame of each clip. Place a "stop()" action in the same frame on the actions layer. Give each movie clip sequence a relevant instance name – if the movie clip name is Dancing Man call the instance name "dancingMan".



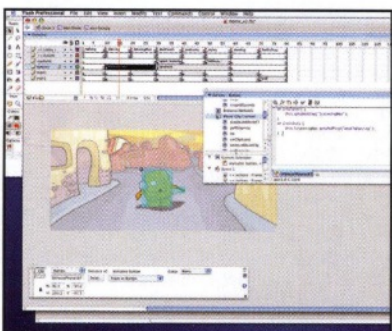
19 Drag the sandwich symbol from the library and make it into a movie clip called "draggingSandwich". Place an invisible button on a separate layer. This will contain the action script that will allow the user to drag the sandwich around the screen and interact with the character.



20 Most of the sandwich man's actions will be contained within an invisible button. To create this, draw a white square and convert it into a graphic symbol called "White Square". Transform this into a Button symbol called "Invisible Button" and place the square in the hit state only.

You said what?

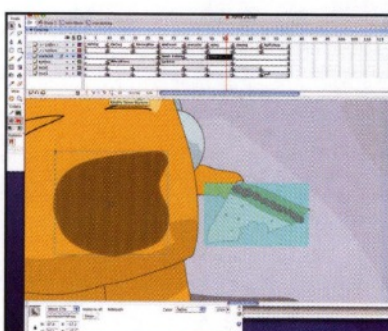
It's worth taking some time to add `//` comments throughout your action script. Your scripting may make perfect sense to you when you are writing it, but come back to the same script a few weeks later and you may end up spending time trying to understand why on earth you wrote it that way. Making comments will save you time and make your file much easier for someone else to understand.



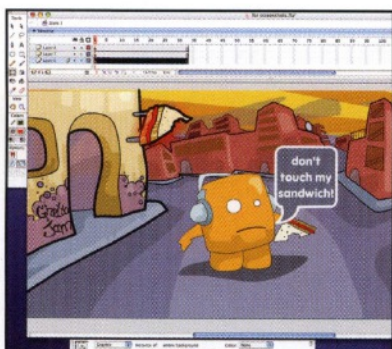
21 To make the character lift his headphones when rolled over, drag an instance of the invisible button onto a new layer in the sandwich man timeline. Add actions telling it to go to the "listeningMan" label when the users rolls over it and back to the "dancingMan" label when they roll out.



22 For varied interactivity you will need a series of actions attached to each of the sandwich's instances. For example, the sandwich has to know what happens when the user rolls over, rolls out, presses, starts dragging and stops dragging or collides with another movie clip.



23 The actions for the sandwich have a collision test, which allows you to test if the sandwich has been released in the mouth or hand. Both the mouth and hand will need a hit area to detect the collision. These hit areas are created from invisible buttons converted into movie clips.



24 Add speech bubbles to help the user interact with the characters. When you take the sandwich, a speech bubble should say, "I want my sandwich", prompting the user to give it back. The speech bubble should be a movie clip that plays as soon as the character starts to cry.



25 Once the sandwich man has started to interact with his sandwich, try to introduce some props of your own. Add other characters to the scene and make use of his other body parts. You could also put into practice the skills you have learned while creating character-based games.



Flash MX: Character animation tricks



All the components needed to complete this tutorial can be found on the cover disc.

Fancy a quick, fuss-free whiz through character animation in Flash? **Jon Burgerman** shows how to animate this brilliant strip, revealing the perfect starter tips to create maximum movement, with minimum effort...

Flash can be used to pull off some brilliantly complex tricks, interactive animations laced with fancy Action Scripting and dynamically loading thingies. Great if you love code and mathematical conundrums. But what about if you just want to use it to give simple animations life and character?

Animation generally requires an abundance of time and patience, which is

unfortunate if you lack both. So we'll show you how to get the best from the doodle, a basic knowledge of Flash and little effort.

The simple tweening of a range of small elements creates the illusion of some serious hard work here. But the fundamentals covered here will give you an understanding of how you would be able to take things further if you wished. You'll create an animated character, the skateboarding

Sparrowbrain (there's a reason for the skateboard), and a scrolling backdrop for him to move along. This backdrop makes clever use of several layers, speeds and changes of direction to create the convincing illusion of sideways movement that is key to animation.

Finally, you'll be able to link these animated scenes together with a cool, but deceptively easy, text mask that effectively introduces your scene. ■

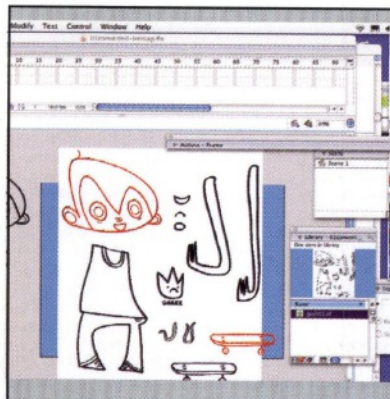


Flash animation

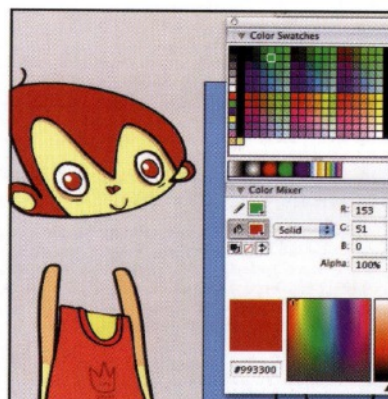
First think about how you want your character to move...



1 It's a good idea to sketch a few rough drawings of the character before hitting Flash – this saves time and avoids that horrible blank screen scenario. Think about the character's movement, and exactly which parts will need animating.



2 Now draw your character in Flash. But if this feels too basic, try using Freehand or Illustrator and then export it for Flash use. You could import a bitmap image (a scan of your image) into Flash and trace over it, locking the bitmap image in a layer and tracing in a new layer above it. You can then convert your bitmap into a vector by choosing **Modify > TraceBitmap**.



3 Add colour to the character using the Paint Bucket tool. The Color Mixer (Window > ColorMixer) can be used to generate tints and shades to add depth, but do bear in mind the environment you'll be placing the character into so that the colours do not clash. ➤

Guides rule

When the movie Stage gets really busy it's sometimes useful to add Guides. To open the Rulers, click View > Rulers and then create a Guide by clicking and dragging from a side ruler on to the Stage. You can now view through clutter by choosing Show All Layers As Outlines. You can then have this toggled on and off on individual layers.

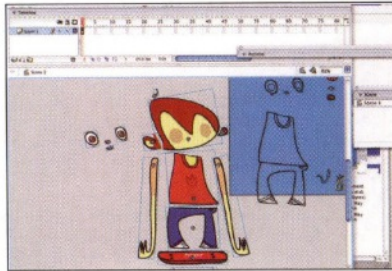


Flash animation

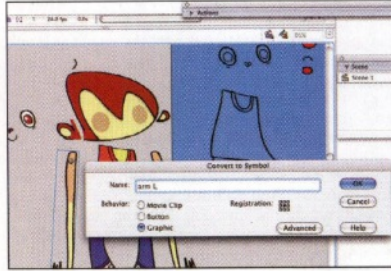
continued

Time saver

Once you've created a Graphic, it can be re-used as many times as you like within your movie without increasing the overall size of the final .swf file. A good tip for arms and eyes, which might look the same for both left and right, is to just flip the Graphic over (Modify > Transform > Flip Horizontal). A frowning mouth can become a smile by rotating it. The same Graphic can then be placed into individual Movie Clips and manipulated without altering the original.



4 Now break your character down into smaller sections so you can manipulate them independently. The dissection of your character can get a little messy so it's a good idea to create a few duplicates first. Copy and Paste the character onto the Stage a couple of times as backups.



5 The separate sections (or Symbols) will form the animated elements of your character. But first, everything needs to become a graphic. Select a body part, the left arm for example, highlight it and click Insert > Convert To Symbol. Make sure Graphic Behavior is selected and name the graphic "arm L". Do the same thing to each animatable part and ensure that they are clearly labelled.



6 Open the Library by choosing F11 or Window > Library, create a folder for the character and deposit all the body parts into it. This will keep things organised, especially when moving the character into other Flash movies. As new elements are added to your Flash file, the Library should be tidied up.



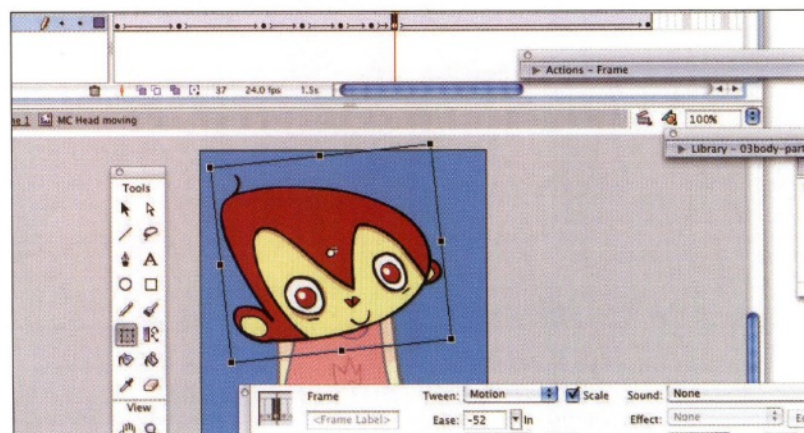
7 Reassemble the character on the Stage with its native body parts in the correct position. Each element must be converted again to a Movie Clip. Select a graphic, click Insert > Convert To Symbol (or F8) and make sure the Movie Clip Behavior is selected. It's a good idea to give them the same name but with the prefix "MC" eg. "MC arm L".



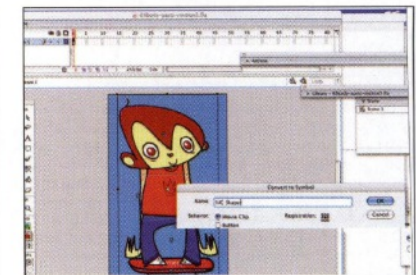
8 The character can now be tweened. On the Stage, select its left arm and double click. There will be a single arm graphic on the first frame of the Timeline. Select this Keyframe and click Insert > Create Motion Tween. Add another Keyframe further along, clicking on it and choosing Insert > Keyframe.



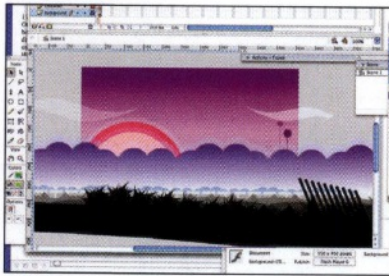
9 The arm Graphic will now appear in this second Keyframe. By manipulating it with the Free Transform tool, Flash will create motion between the two frames. Hit Enter and watch the animation. You can rotate it, resize it and change its visual properties by using the Properties toolbar (Window > Properties). You can add more Keyframes along the Timeline, too.



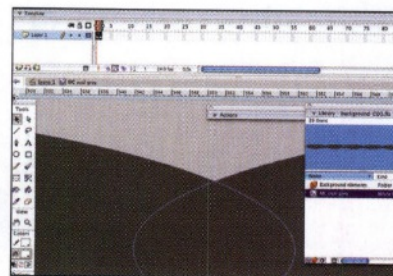
10 Click anywhere between two motion tween Keyframes to adjust the Ease of the transition using the Properties toolbar. Try tweening the other Graphics. Not all of the motions need to be exaggerated; subtle movements can be just as effective.



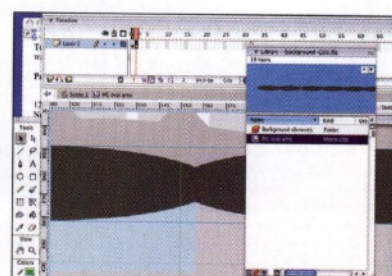
11 Once each element has been tweened, the character can be saved as a Movie Clip of its own. If depositing the character into different Movies, you could save different Movie Clips of your character waving, asleep or angry. Highlight the clips and select Insert > Convert To Symbol to make it a total Movie Clip. Alternatively you could save it as a Graphic and then a Movie Clip, edit the Movie Clip and Tween the whole Graphic.



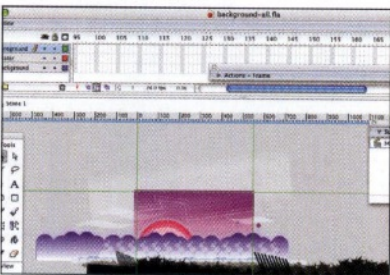
12 You now need to create an environment for your character. Backgrounds do not have to be fully animated, although elements such as fires, flickering lights or rain can help create a strong atmosphere. You can now knock out a quick scrolling background. Open the file background-static fla from the cover disc to see how these layers have been set up.



13 Hide the Foreground layer by clicking the first dot by its name, click on the Graphic "oval grey" and convert it to a Movie Clip named "MC oval grey". Double-click on it and create a Motion Tween on the first frame. Put another Keyframe in the next frame. Turn on Onion Skin Outlines. Move "oval grey" until it lines up with the Graphic in the first frame.



14 Drag the second Keyframe further out along the Timeline. Hit Enter and the ovals should animate, moving to the left. Check the file background-oval fla on the disc and compare it to yours. Also included on this file is a more complicated background object so you can see how the same idea can work for any shape, as long as it's repeated.



15 Apply the same principles to both the foreground and background elements, remembering that those in the foreground should move faster. Stretch out the distance between Keyframes to make the Graphic move more slowly. To create parallax scrolling effects, as seen in side scrolling computer games, experiment with moving some background elements the opposite way. Look at the executable cover disc file "background-all" to see how it can be built up.



16 Copy and paste your character into the layer marked Character. To preview, click Control > Test Movie. Try to keep the frame rates of your movies the same or you'll have problems when bringing objects in from other movies. I made Sparrowbrain a skater so that his legs didn't move – a sneaky way to cut down on work!



17 This swirling text mask will appear as a solid object that becomes a window through which the movie will appear – an alternative to Flash's, sometimes jumpy, layer Mask. Firstly get some text onto the Stage and convert it to a Graphic called "text".



18 Convert to a Movie Clip ("MC text") and turn the first frame into a Motion Tween. Create a second Keyframe further along. Return to the first, make the text graphic small and rotate it. Add a third (large text, right way up), to look like text-tween fla.



19 Draw a white square in the main Stage and convert it to a Graphic ("square white"). Delete it from the Stage and double click "MC text" and "text" in the first frame. Select all and click Modify > Group. Drag the Graphic "square white" onto the Stage, stretch to fit and place it behind the Graphic "text". Select All, click Ctrl+B and delete the text to create the CD file text-tween-mask fla.



20 Copy and Paste the mask "MC text" into a layer in the Background movie. The Mask plays for 40 Frames, the length of the animation. The animation beneath comes in on Frame 21, the point where the mask zooms out. Play with the duration of your Movie Clips to get the optimum timing. Later, these easy techniques can be utilised to add a bit of fun to your web site or presentation.

Quick copies

To make a quick duplicate of a Movie Clip or Graphic, open the Library, select what you want to copy and Right-Click (PC) or Ctrl-Click (Mac) and select Duplicate. Rename the symbol you are duplicating and it will appear in the Library. The symbol can then be edited via the Library or by dragging it out onto the Stage and double clicking on it.

Importing into Flash

Integration between Fireworks and Flash has become much easier. Converting a Fireworks file is now as simple as opening it in Flash. A small dialogue box will present you with options for the file, object and text structures. Or you can paste single or multiple objects from the clipboard. But beware: though transparency and gradients are supported, many textured or filtered objects will convert into transparent bitmaps.



Illustration & tutorial by Ailin Chambers

Flash MX: Lip & facial synching



All the components needed to complete this tutorial can be found on the cover disc.

Although lip-synching and character animation are a crucial part of successful animation, it takes skill to make your character move and talk convincingly. Flash provides simple and effective methods of achieving great results...

In any form of animation, the principles are the same. But technology increasingly enables you to expand your animation into many different mediums — and Flash MX, of course, is no exception.

Flash provides a quick way of animating 2D images for use in both on the web and television. However, its creative potential should never be underestimated; the power it can provide to you, the animator, is massive.

It goes without saying that character animation is important, especially when the character itself is talking. Gestures and lip-sync work together to give your character life and personality. It's often easy to forget that the animator is the actor and so it's important to ensure that the character acts, too. Facial expressions and subtle head movements will make your character less wooden.

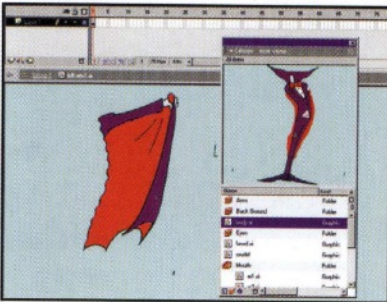
Flash provides a simple and easy way to construct characters, whether they're drawn

directly onto your computer or on paper first. The following tutorial will introduce you to those methods and to the creative process behind making an animated character talk. By concentrating on movement, lip-synching and a range of Flash tools, you'll ensure that your character really does come to life.

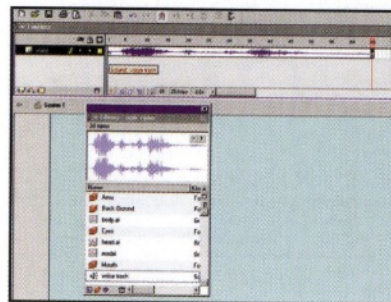
So have a go with the examples provided on the cover disc and then see what effect the skills you learn have on your future animation projects. ■

Animating in Flash

It's a good idea to break up your character as much as you can...



1 Keep eyes, eyebrows and mouths separate from the head, and arms and legs separate from the body. Put their symbols into Library folders so they're easy to find.



2 Drag the Library sound file to the timeline. To determine your scene's length, click on a frame on the voice layer and hold down F5 until the entire track is visible.



3 Bring the character into the main stage layer by layer. At this point, leave the head empty. It's also a good idea to label your layers, to prevent confusion later on.

Onion skin

When moving symbols in Flash to animate a movement along the timeline, it's often tricky to judge where the previous frame was on the stage. This is where the Onion Skin comes in handy. It acts as a virtual light box, showing semi-transparent images of the frames before and after the frame you're currently on. This is very helpful when animating cycles like walking or birds flying.



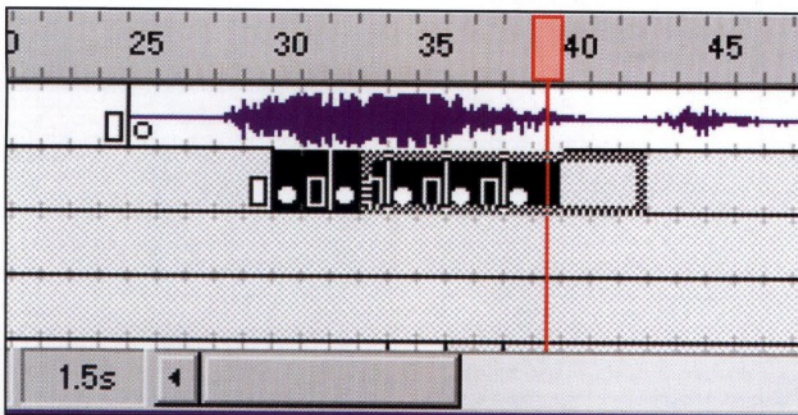
4 If you have drawn your animation on paper, draw crosshairs on each drawing so that the animation can be accurately matched up again once it's been scanned and imported into Flash MX.



5 Now to assemble your character. In this case, the vampire moves his arms up as he speaks. Because the body remains still, it stays as a single frame throughout the movie. Place the arms on separate levels, one layer above and one layer below the body.



6 Listen to the dialogue and pick out words that you could emphasise with some movement. Tying this to a particular word or phrase can really add life to a character. Here, for example, our vampire raises his arms as he says, "Welcome!"

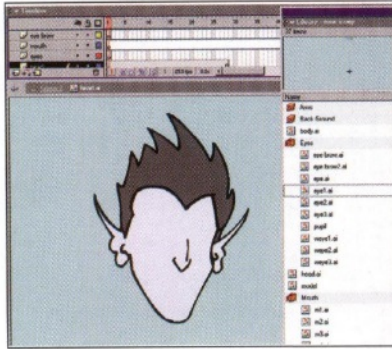


7 Play the animation back or scroll along the timeline. If the movement doesn't seem to tie in properly with the dialogue, highlight the frames of the animation and drag them left or right along the timeline to improve the sync. Once you're happy with the results, move on to step 8.

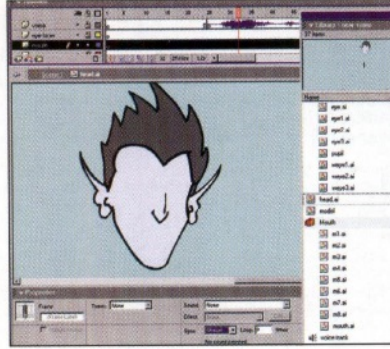


Animating in Flash

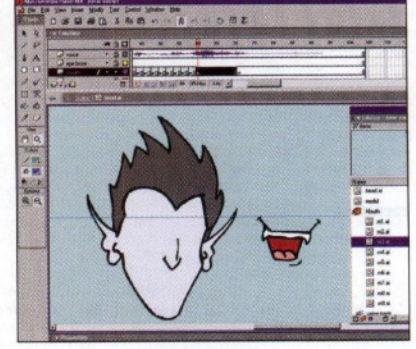
continued



8 Double-click the head symbol and then start to bring other symbols onto it. Next, create layers for the eyes, eyebrows and mouth. By animating these elements in the head symbol, you will be able to move it about on the main stage without worrying about repositioning the features.



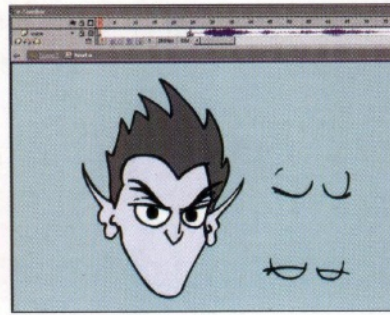
9 When lip-syncing within the head symbol, bring in the sound file as you did in the main stage. But before you drag the mouth shapes from the library onto the head, make sure the mouth layer is selected and remember to set the sound object in the Properties box to "stream".



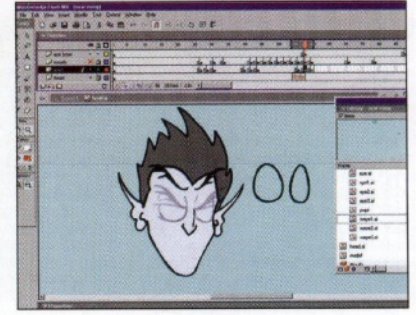
10 There are basically nine mouth shapes for lip sync. Listen to the dialogue on the timeline and pick out the phonetics and the accents. On the mouth layer, create a blank frame (F7) and bring the relevant mouth position onto the stage from the Library.



11 Your character's blinking and eye expressions can also help to give the dialogue more expression. Think about what your character might be doing while talking. Is there an expression in the voice that can be given extra effect by using a particular eye or eyebrow movement?



12 Whatever expressions you decide to use, always remember to make your character blink, as this will help to make it look less wooden. There are many different ways to animate blinks, but generally three drawings make up the movement.



13 Blinks are particularly useful when you need to seamlessly slip from one expression to another. Start with the eyes fully open, then fully closed, then finally half open. From here, you can either go back to the original (fully open) position or on to a new eye shape, such as wide or narrow.



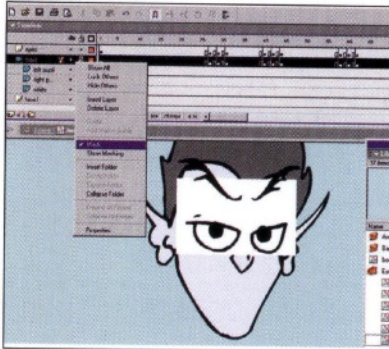
14 During this change of expression, you'll need to animate the eyebrows in order to cushion their movement from the first expression to the next. Onion Skin the frames of the eyebrows. Create a keyframe (F6) between the two frames and move the new eyebrow frame in between the two.



15 Position the character's pupils on separate layers, directly beneath the eye layer, so that they are free to move around without affecting the blinks and eye shapes. Add one more layer underneath these, label it "white", and draw a white rectangle that covers all the eye positions.



16 Because the pupils are now on a separate layer, you need to create a mask layer to hide the parts of the pupils and whites that overlap the eyes or eyelids. Start by animating all the blinks and expressions you need for the scene, then highlight all frames in the eye layer and Copy them.



17 Create a new layer and Paste the eyes onto it. Name this "mask". Right-click this new layer and select "mask" from the dropdown. This layer is now blue. Drag the layers to be masked onto it to make them blue, too (if they're not already).



18 Now highlight each frame on the mask layer and in turn colour the inside of the eyes. In this example, they're red. To do this, you need to break the symbols apart by pressing Ctrl+B. Lock all the timeline layers to see it work.



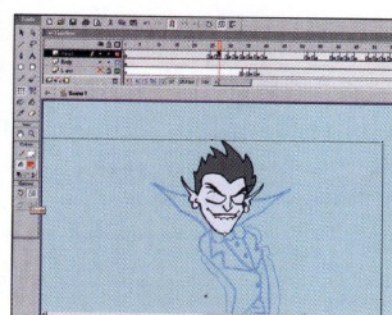
19 It is often a really nice addition to dip and raise the head when the character is talking. Again, listen to the words and emphasise the expressions. If it helps, match the head movements with the eye expressions. Do this sparingly to prevent your character from jittering.



20 Start by plotting keyframes on the head layer. Add a frame at the point you intend to move the head. Highlight each of these frames and adjust its position. As the vampire begins to say "Welcome!" the head dips. As he finishes the word, the head rises.



21 To avoid making the head too snappy, add more keyframes between the newly created head positions. Now move them between to cushion the head movement. Use the Onion Skin feature to help see the frame before and after.



22 Give the head a final tweak by adding "squash and stretch". At the head's lowest point during a movement, use the Free Transform tool (Q) to squash it down and widen it. Do the opposite for the head's highest point. Don't overdo this; be subtle.



23 Create a layer below the character in the main timeline and bring in the background (either create this in Flash or import it from your computer). Drag the background onto the background layer.



24 Go through the symbols and remove any markings and crosshairs with the Eraser (E). Now's the time to make any final adjustments... Happy? Then hit Ctrl+Enter to Preview your movie or Export it as an AVI file.



25 Lip-synching and character animation in Flash needn't be complicated. Keep things simple and don't move anything unnecessarily. Play with other Flash tools and see what you can create.

Masking

Masking is an often underused tool that gives a lot of freedom to a scene. The mask always covers the area you want to see in the movie, leaving the surrounding areas hidden. For example, if a character walks behind a lamppost, which is part of the painted background, on the mask layer, every area is painted except the post itself.

Animating a symbol

Cycles are often used to animate inside a symbol, enabling that symbol to be moved around later in a scene without adjusting the animation. If you have all your animation contained in one symbol, then you can move it around in the main timeline, using Motion Tweening to make it move across the screen, or get larger or smaller.



Studio MX: Countdown clock

THE FINAL DESIGN



All the components needed to complete this tutorial are on your cover disc.

Fire up Flash and Fireworks to create a cracking countdown clock with **Tom Knowles**

In the current climate of usability, accessibility, code standards and other po-faced aspects of web design, it's easy to negate the need for experimentation.

Yet, some of the best web site concepts were born out of going down blind alleys in the first place. And while the current focus on usability — and all that — has to be a good thing, there are still some good opportunities

for you to stretch the boundaries every now and then.

So if you're asked to do something different or build something temporary, akin to a flyer or an advert, then it pays to think outside the usual parameters. Notable developers who have used these sorts of projects to push the industry's expectations of Flash include Joshua Davis (www.joshuadavis.com), Branden Hall (www.waxpraxis.org) and Yugo Nakamura (www.yugop.com).

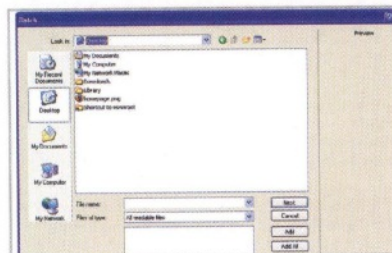
Here, we're going to produce an interesting holding page for a forthcoming bar. Rather than going down the usual under-construction-style HTML layout, we're going to create a Flash clock to count down to the opening date, and which uses an XML-driven selection of images in a random and eye-catching way. ■

Part 1: The countdown clock

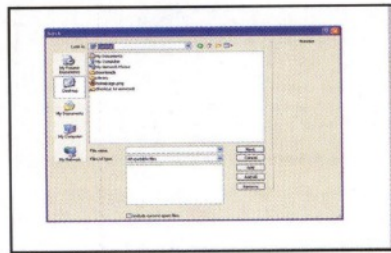
How to prepare the images for the countdown clock

Improve performance

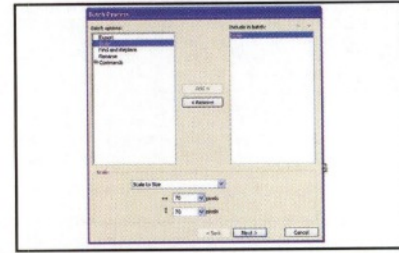
Macromedia is trying to improve performance for the Flash player, but for now, don't over-saturate your work with complex animations and heavy graphics. It can cause performance and playback problems — more so with fast-moving loops.



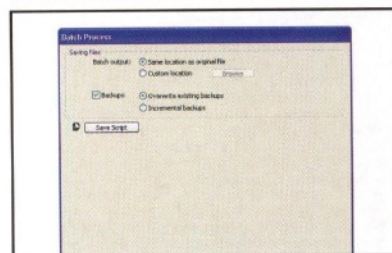
1 Our countdown clock will feature a border of randomly cycled image thumbnails. Rather than manually resize each image down to the correct size, we'll use Fireworks MX 2004's batch process function. Open Fireworks, and choose File Batch Process. Once you've done this, you'll be presented with the explorer box.



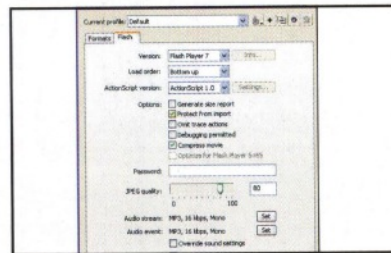
2 Browse to the folder where you're keeping all the full-size images, and select all the photos you need. For this project, we're going to need 20 images. Once you've got your selection, click Add, followed by Next. You'll now be given a number of processing options. Choose Scale and hit the Add button.



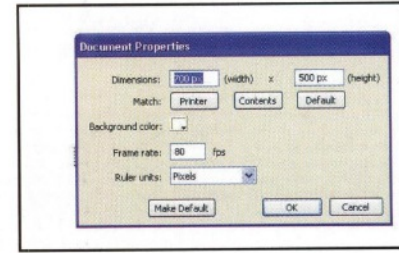
3 In the drop-down at the bottom of the window, pick Scale to Size, and set the size as 70x70 pixels. Now return to the Batch options section and choose Export. This time, pick JPEG — Smaller File from the drop-down. You can edit the precise export settings if you wish. When you're done, click Next.



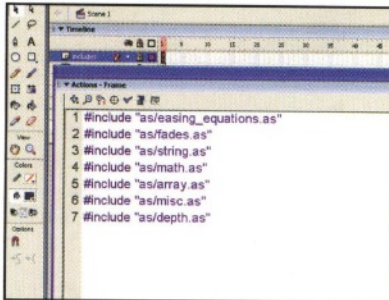
4 In the next screen, pick a new folder for your soon-to-be-resized images. We've called our folder thumbs. Now, simply hit Batch and sit back and relax. Fireworks will trawl through the original list of images, open each one and resize to your specifications, before exporting it to the new folder. Cool huh?



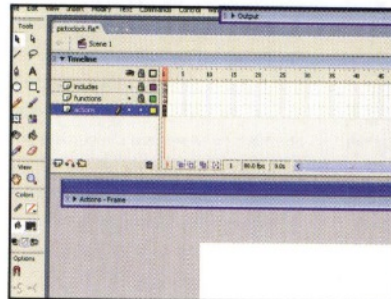
5 That's all we need to do in Fireworks, so now let's open Flash MX 2004, and create a new file. Save the file as pictoclock fla, and then choose File > Publish Settings, and hit the Flash tab. Set the ActionScript version to ActionScript 1.0, and check the Protect From Import and Compress Movie boxes.



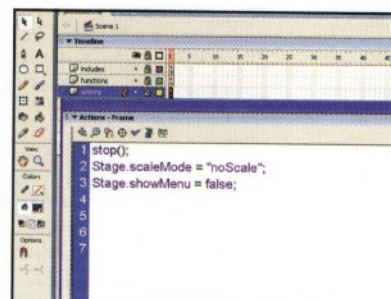
6 In the Document Properties panel, set the stage dimensions to 700x500 pixels, and the frame rate to 80fps. The high frame rate ensures that we can evaluate the time quicker than the real time clock. The document size is reasonably unimportant, since we won't be scaling the movie, but we will use the values in our code.



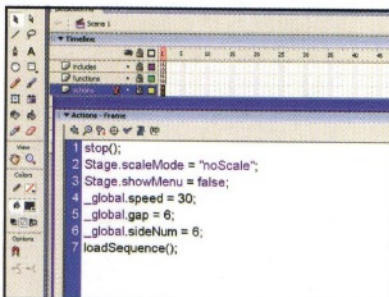
7 On the main timeline, rename the first layer to includes. In the Actions panel (Window > Actions), type the above code. These include statements refer to external ActionScript files, containing functions and prototype methods. The easing_equations.as file contains Penner's easing equations (www.robertpenner.com).



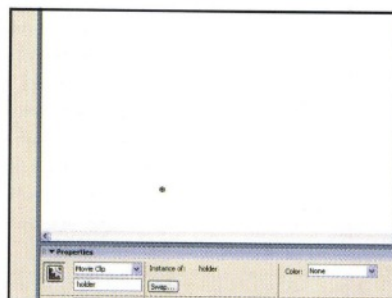
8 Next, create another layer beneath includes, and call it functions. Leave this layer blank for now – we'll return to it later when we write our main ActionScript code. Underneath it, add another layer called actions. This layer is where we'll put a few brief AS statements which deal with the initialisation of the overall movie.



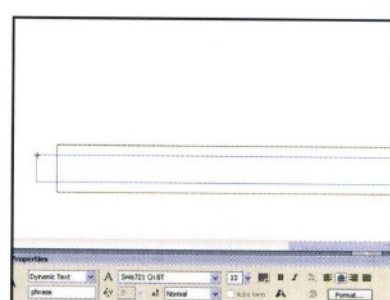
9 Type the above code in the Actions panel. The stop(); line halts the movie's playhead, preventing looping. The next two lines make use of the Stage object – we remove the user's right-click options, and noScale prevents the movie from being scaled or distorted. This is important since we'll be taking up 100% of the browser window.



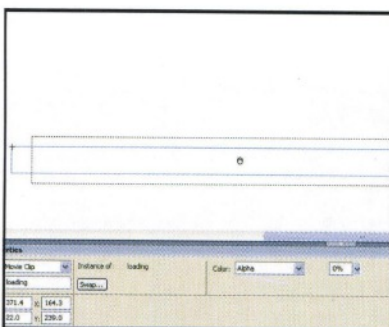
10 The global variables, _global.speed, _global.gap and _global.sideNum refer to the fade transition speed, the gap between the thumbnails and the amount of images on each border's side. Using the _global prefix, we can access these from anywhere. We call the loadSequence function, which manages the data load.



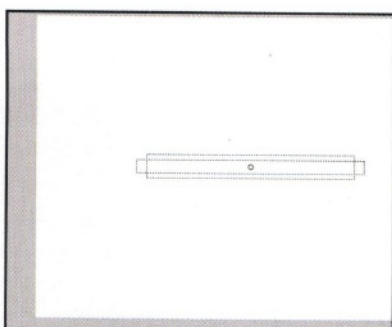
11 Now create a new layer called images, and choose Insert > New Symbol > MovieClip, giving it the name holder. This is essentially just a blank MovieClip, so all we need to do is drag it onto the stage, and give it the instance name holder. You don't have to worry about its position for the moment.



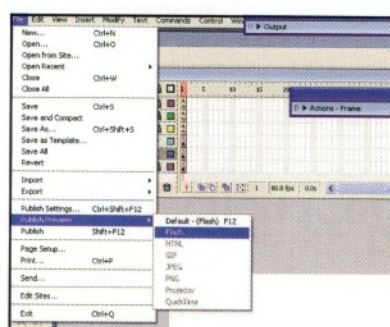
12 On yet another new layer (it makes sense to keep everything on separate layers for ease of editing and organisation), create a new text field. In the properties panel, set it to dynamic text, and give it the instance name of phrase. Centre-align it, and don't forget to embed the font in the character options box.



13 Select the phrase textfield, and choose Modify > Convert to Symbol > MovieClip. Give this new symbol the name and instance name loading. On the main stage, use the Align panel to centre it vertically and horizontally. This MovieClip will display the load progress of the movie, the XML feed and the thumbnail images.



14 Create another new text field (with the instance name timefield) on a new layer, and place it inside a MovieClip with the instance name timeholder. Make the font size around 40. Again, on the main stage, centre the clip horizontally and vertically. Finally, set both the timeholder and the loading MovieClips _alpha values to 0.



15 That's all the pointing and clicking we need to do. The rest of the countdown clock's functionality will be produced purely by the ActionScript code on the next couple of pages. When you've entered all of the following code, simply choose File > Publish Preview > Flash to test the final movie.

Take control with CS

Fireworks' batch processing can save time and effort, but when it comes to real, in-depth control, you can't beat Photoshop CS. You can record any sequence of commands, save them as a script and then apply that script whenever you wish.



Part 2: Init(); initialization, & loadSequence() functions

The complicated bit...

Whenever you build anything reasonably complex, the structure of your code is just as important as the structure of your .FLA file. By structure, we mean the breakdown of your script into functions, and the logic of its architecture. Basically, if you've got one function that is 100 lines long, you need to think about restructuring it. The reason for this is essentially ease of debugging. By breaking your code down into the smallest chunks possible, you not only make it easier to understand, but you also make it quicker to search for bugs, as you're able to narrow down the areas where something might be going wrong.

A classic structure for any application is to have an initializing function, which kick-starts proceedings, and then lets the rest of the application get on with its various jobs. In the following code, we cover this init(); function in addition to the loading script..

1 Lines 1–13: The very first thing we do in our init() function is to declare the date we're actually counting down to. We do this by creating a new Date object (countdownYear), and stuffing it with the date we want. Take note that the way we actually provide data for the Date object is a little strange, what with Flash treating the first month of the year as 0 rather than 1. Once we've done that, we work out exactly how much time there is between now and then by using a Date object that corresponds to the current day.

```
function init() {
    _global.countdownYear = new
    Date(2005, 1, 2, 0, 0, 0, 0);
    today = new Date();
    var millidif = countdownYear.
    valueOf() - today.valueOf();
    days = Math.floor
    (millidif/86400000);
    millidif -= (days*86400000);
    hours = Math.floor(millidif
    /(86400000/24));
    millidif -=
    (hours*(86400000/24));
    minutes = Math.floor
    (millidif/(86400000/24/60));
    millidif -= (minutes*
    (86400000/24/60));
    seconds = Math.floor(milli
    dif/(86400000/24/60/60));
    millidif -= (seconds*
    (86400000/24/60/60));
    hunseconds = Math.floor(milli
    dif/(86400000/24/60/60/100)
    ); var start = getTimer();
```

2 Lines 14–18: Since we're trying to display the time left in the form of a digital-clock readout, we need to do some crafty formatting of the numbers before we display them. To do this, we use a custom function (in the as folder) that formats the numbers to display as double-digits..

```
daysleft = simpleFormat
(_root.days);
hoursleft = simpleFormat
(_root.hours);
minutesleft = simpleFormat
(_root.minutes);
secondsleft = simpleFormat
(_root.seconds);
hunsecondsleft =
simpleFormat(_root.hunseconds);
```

3 Lines 19–27: The meat of the clock itself is this onEnterFrame statement. It runs at the frame rate of the movie, and each time it checks the elapsed time (with the getTimer() method) to see what unit of time has passed. If less than a second has passed, we skip the rest of the function. If not, we further analyze the current time..

```
root.onEnterFrame =
function() {
    elapsed = getTimer() -
    start;
    if (elapsed < 1000) {
        continue
    }
    else {
        start = getTimer();
        timeholder.
        fade(100,speed*3);
        hunsecondsleft -= 100;
```

4 Lines 28–46: We break down the elapsed time into hundredths of a second, seconds, minutes and hours. After doing that, we can then see which values need to change. Eg. If the minutes left are zero, we reset them to 60 and decrement the hours left. If the hours left are zero, we decrement the days left and reset the hours back to 24.

```
if(hunsecondsleft < 0) {
    updateClock();
    hunsecondsleft += 100;
    secondsleft -= 1;
}
if(secondsleft < 0) {
    secondsleft += 60;
    minutesleft -= 1;
}
if(minutesleft < 0) {
    minutesleft += 60;
    hoursleft -= 1;
}
if(hoursleft < 0) {
    hoursleft += 24;
    daysleft -= 1;
}
timeholder.timefield.
text = simpleFormat
(daysleft) + ":" +
simpleFormat(hoursleft)
+ ":" + simpleFormat
(minutesleft) + ":" +
simpleFormat(secondsleft);
updateAfterEvent();
}
```

```
holder.fade(100,speed*3);
}
```

5 Lines 47–62: The loadSequence function first loads the site itself, using the usual getBytesLoaded/getBytesTotal routine. Once the site is loaded, we then create an XML object (thumbXML), and load our XML file into it. Notice how we provide information to the user throughout, via the loading text field. When the XML feed is fully loaded, we call the processXML function.

```
function loadSequence(){
    loading._alpha = 100;
    loading.phrase.text =
    "LOADING SYSTEM";
    _root.onEnterFrame =
    function() {
        if( _root.getBytesLoaded
        () == _root.getBytesTotal
        () ) {
            loading.phrase.text =
            "LOADING DATA";
            _global.thumbXML =
            new XML();
            thumbXML.ignoreWhite =
            true;
            thumbXML.onLoad =
            function() {
                loading.phrase.text
                = "LOADING IMAGES";
                processXML();
            };
            var xmlpath = "xml/
            thumbs.xml";
            thumbXML.load(xmlpath);
            delete _root.
            onEnterFrame;
        }}
    }
```


Part 3: Finishing touches

UpdateClock(), processXML() & the arrangeImages() functions...

The backbone of the countdown clock is the array of thumbnail images it displays. As with any Flash application worth its salt, our clock fetches its images via an XML file (included on the cover disc), using the classic For Loop to iterate over the structure and stuff the images into a global array. To actually load the images themselves, we use dynamically created instances of the MovieClipLoader class. Once we've got our images, we use a general function called arrangeImages, which places the images (in random order) in a square layout. What's useful about this is that we can use arrangeImages whenever we like, whether it's every second, every minute, or whenever the user clicks their mouse. By encapsulating the arrangement functionality within a separate function, we've instantly allowed ourselves flexibility that we wouldn't have had if we'd kept all of the code in one large chunk.

1 Lines 1–3: This function, updateClock, is called every real second. Notice how it doesn't do an awful lot, apart from calling other functions. You may notice, however, that it calls a method on the imgArray called 'mix'. This is actually a prototype method, declared in one of our external ActionScript files. A reasonably simple but very useful method, it randomly jumbles the order of a given array. With the imgArray jumbled, we can then call the arrangeImages function, which loops through it, placing the image.

```
function updateClock() {
    imgArray.mix();
    arrangeImages();
}
```

2 Lines 4–8: The processXML function takes the thumbXML data structure, and parses it, pulling out the thumbnail image paths and stuffing duplicated MovieClips into the imgArray. Notice how we use child/parent syntax when dealing with XML documents..

```
function processXML() {
    var thumbs = thumbXML.
    firstChild.childNodes;
    _global.imgArray =
    new Array();
    _global.loadedThumbs = 0;
    for (var i=0; i<thumbs.length;
    i++) {
```

3 Lines 9–14: The duplicateMovieClip statement itself. We use the eval() statement to create a concise, temporary reference to the current clip. This is a more efficient way of handling dynamically created MovieClips, because we don't have to keep using the same complex reference syntax, and hence leaving ourselves open to typos.

```
holder.createEmptyMovie
Clip("img"+i,i);
var temp = eval("holder.
img"+i);
imgArray.push(temp);
var imgloc = "thumbs/" +
thumbs[i].attributes.path;
holder.createEmptyMovie
Clip("loader"+i,2000+i);
holder._alpha = 0;
```

4 Lines 15–26: To use the MovieClipLoader class properly, you need to pair it with a listener object which listens to

the events it broadcasts, and acts accordingly. One of the most important events is onLoadInit, which is fired when the movie is loaded and initialized. It's important to distinguish between this and onLoad – which is fired before the initialization, meaning that you can't immediately access the movie's properties.

```
var tempLoader =
eval("holder.loader"+i);
tempLoader.mclListener =
new Object();
tempLoader.mclListener.
onLoadInit =
function(target_mc) {
    loadedThumbs++;
    if(loadedThumbs ==
thumbs.length) {
        loading.phrase.text
        = "";
        arrangeImages();
        holder.center();
        init();
        arrangeNav();
    }
};
```

5 Lines 27–36: The beginning of the arrangeImages function creates a couple of variables (xval and yval), a For loop, and a reference to the current element in the imgArray. The element represents an image thumbnail clip, and we set its coordinates to the values of our xval and yval variables.

```
tempLoader.image_mcl =
new MovieClipLoader();
tempLoader.image_mcl.
addListener(tempLoader.
mclListener);
tempLoader.image_mcl.
loadClip(imgloc, temp);
}

function arrangeImages() {
    var xval = 0; var yval = 0;
    for (var i=0; i<imgArray.
length; i++) {
        var temp = imgArray[i];
        temp._x = xval; temp._y = yval;
```

6 Lines 37–56: The last part of this function is the trickiest, because in it we need to look at the current MovieClip's position in the imgArray and decide where in the arrangement it should sit. There four sides to the square, and hence four 'zones'

in which the clips can be, and so it's case of establishing the point at which we need to move to a different zone, and altering the coordinate incrementation to suit.

```
if(i>4 && i< 10) {
    if(i == (sideNum -1)) {
        _global.toprightX =
temp._x;
    } yval += (temp._
height + gap);
}
else if(i> 9 && i< 15) {
    if(i == ((sideNum*2)-
2)) {
        _global.botrighY =
temp._y;
    } xval -= (temp._width
+ gap);
}
else if(i>14) {
    if(i == (sideNum - 3)) {
        _global.botleftY =
temp._y;
    } yval -= (temp._
height + gap);
}
else {
    xval += (temp._width +
gap);
}
}
```




Studio MX: Sliding puzzle

THE FINAL DESIGN



Components to complete this tutorial are on your coverdisc.

More madness from the MX master, **Tom Knowles**. This time, making a sliding puzzle with Freehand, Fireworks and Flash

Macromedia's Freehand application is often said to live in Adobe Illustrator's shadow. It has the same purpose and offers much of the same functionality, but is often dismissed as being a poor relation. There is, of course, some truth to this.

Illustrator is the tool of choice for all the top illustrators and print designers, simply because it's at least one version ahead of

Freehand, and has been around for longer.

Freehand does have a trump card, however, and that is its seamless integration with other tools in Macromedia's Studio MX suite. Despite an increase in effort from Adobe, trying to work between Illustrator, Flash and Fireworks is a nightmare of functionality loss, colour conversion issues and layer compression. Substitute Freehand for Illustrator, though, and you'll be able

switch between apps, cutting, pasting and editing seamlessly.

We'll use Freehand to create a vector illustration, suitable for web or print usage, and then cut it up into a grid with Fireworks.

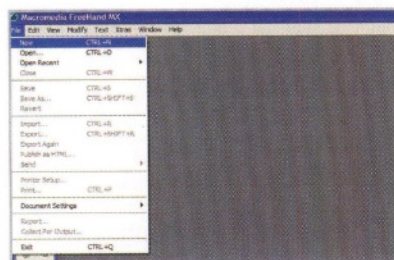
Finally, we're going to use Flash to create a classic game. ■

Part 1: Create, export and animate your game character

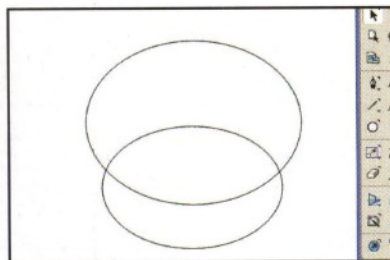
From the initial illustration...

Avoid trial and error

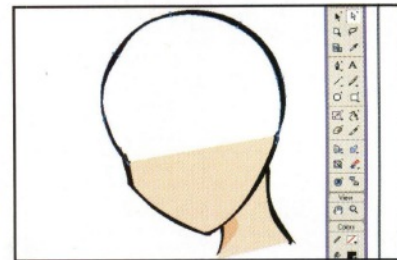
When approaching ActionScript gaming projects, it's important to plan your code structure, rather than attempting to produce your game via trial and error. Start with loose, informal notes, and evolve them into pseudo-code.



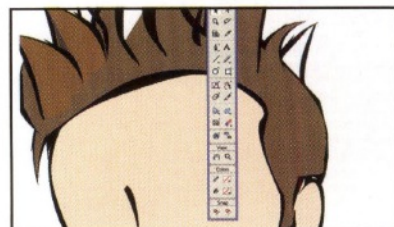
1 Okay, let's deal, firstly, with our initial illustration. Open Freehand MX and create a new document by clicking File > New. Don't worry about changing the document size, but save your file somewhere before we progress. If you're used to using other Studio MX applications, Freehand's layout will seem familiar.



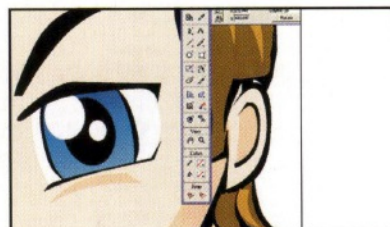
2 Our illustration is going to be a Japanese animé-style character. As with any illustration, the basic shapes and proportions are very important, so the first thing we're going to do is to lay out the shape of the character's head. Start by drawing out a couple of circles with the ellipse tool (accessible via the toolbar).



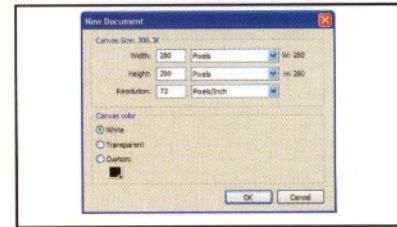
3 Animé characters tend not to have particularly simple shapes, so the best way to get going is to use the ellipse tool, and manipulate its points with the sub-selection tool. This can take some getting used to, but once you become comfortable with the tweaking approach, you'll be surprised at how effective it can be.



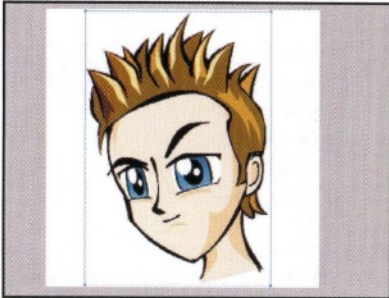
4 By layering shapes on top of one another, you can begin to achieve a sense of depth. For the character's hair, rather than redrawing a similar shape every time, simply group one clump, and copy and paste it multiple times, modifying its proportions for variety. For advanced modifying, choose Window > Transform.



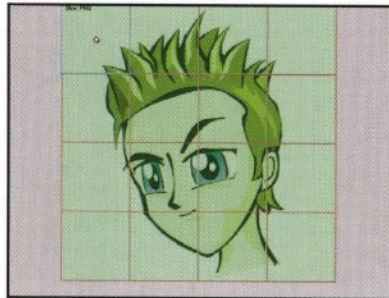
5 One of the most difficult things about animé characters is their eyes. Getting them right takes practice. Luckily, there are loads of web sites dedicated to this style of drawing and a quick Google search will provide you with plenty of help. Once you've got a decent eye, however, you can reuse it over and over again.



6 When you're happy with your illustration (make sure to use highlights and shadows, they can make all the difference), save your Freehand file, and open Fireworks MX 2004. Choose File > New and set your document size to 280x280 pixels at a resolution of 72 dpi, with a white background.



7 Now choose File > Import and browse to your Freehand document. Select it and then, keeping the default options, place it on the stage. Align your image to the centre of the document (Window > Align). Using the line tool, create a four by four grid over the image, ensuring that each square is 70x70 pixels in size.



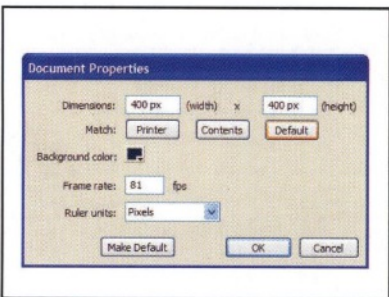
8 Next, pick the slice tool, and create 16 slices, corresponding to the grid. Once you're done, save your file, and choose File > Export Preview. On the next screen, choose to export your file as PNG-8, and click export. You'll be given the option of exporting slices, select this and then hit save.



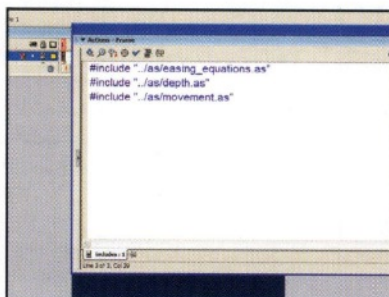
9 Once you've done that, browse to the folder where you chose to export your slices. If all has gone to plan, you should see 16 image slices, their names will be made up of their row and column positions. We'll make use of these names later on when we use the images in Flash.

For print or screen?

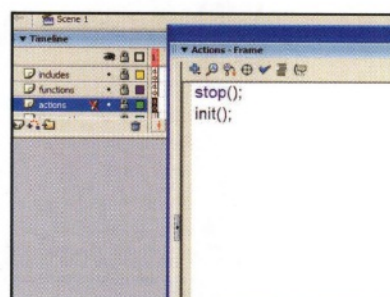
Should you wish to use Freehand to design for print rather than the screen, remember that you'll need to switch to using CMYK colours in the mixing panel, rather than RGB. Beware, CMYK colours can look dramatically different on screen.



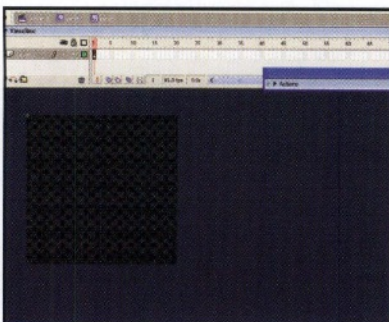
10 Okay, it's time to open Flash MX 2004. Choose File > New > Flash Document, and create a document of 400x400 pixels, frame rate 81fps. Save your file as puzzle fla and, in publish settings, choose ActionScript 1.0. Rename the first layer on the time line to Includes, and then open the Actions panel (Window > Actions).



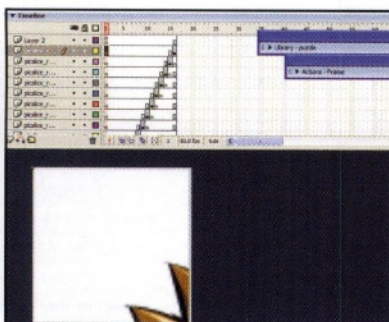
11 Type out the above ActionScript snippet. This imports extra code on the CD, which extends the MovieClip object with new methods, including functionality for movement and depth. Create two more layers, called Functions and Actions. Skip the functions layer – we'll write out our ActionScript there later on.



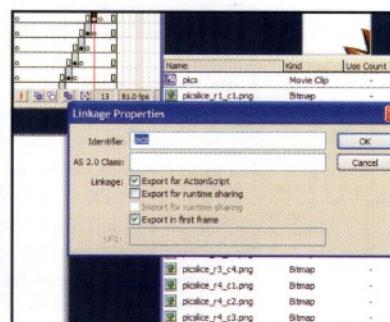
12 On the actions layer, type out the above code. This simply stops the movie playhead (to prevent looping) and calls the init function, which we'll write and explain later on. Next, create two more layers below actions, and call them Pieces and Squares. On both layers, draw a 70x70 pixel square, positioned at 60,60.



13 On the pieces layer, make the square a MovieClip called piece (Modify > Convert To Symbol > MovieClip). Return to the main time line, and give it the instance name Piece. We're going to nest it inside another MovieClip, so convert it to a MovieClip, this time called Pieces. Give this parent the instance name Pieces.



14 Repeat the previous step with the squares layer, this time calling the child clip Square and the parent clip Square. Next, we need to create a new MovieClip called pics. Inside this clip, choose File > Import, and select all the image pieces we sliced up earlier. Once they appear, right-click and choose Distribute to layers.



15 Taking the image names as a cue, move each individual image slice to its own frame, so image_r1_c1.png would be on the first frame, and image_r2_c1.png would be on the fifth, for example. In the library, right-click on pics, and in its linkage properties, choose 'Export in first frame', and give it the identifier Pics.



Part 2: Enabling and disabling the grid functions

Understanding the initialisation and gameplay functions

The init function is good practice when it comes to creating ActionScript applications of all sizes. It is important, however, not to bloat this function – its purpose is to encapsulate the initialisation of the application within an easy, concise set of code – not to do all of the application's work for it.

This goes back to the principle of Object Oriented Programming; whilst we're scripting a fairly simple program and are not really delving into OOP here, it pays to keep the right objectives in mind. First and foremost, that means breaking the application down into the smallest parts possible (or necessary). Therefore, the initialisation element of the program should do nothing more than what it's designed for. By sticking as rigidly as possible to this modus operandi, our code can stay more understandable, manageable and easier to improve.

1 Lines 1–3: To represent the game's board, we're going to use a two dimensional array called grid. Notice how we declare two arrays: rows and columns, before shoving them into our grid.

```
function init() {
    _global.rows =
    new Array(4);
    _global.columns =
    new Array(4);
    _global.grid =
    new Array(rows, columns);
```

2 Lines 4–10: To assemble our grid, we employ the trusty duplicateMovieClip command, and use it to create two layers of blocks – one to register the user's clicks, and one to actually display the movement of the pieces. We nest one For loop inside another, enabling us to build the grid row by row.

```
counter = 0; var xval =
0; var yval = 0;
for(var rowItr =
0; rowItr <
rows.length; rowItr++){
    for(var colItr =
0; colItr < columns.
length; colItr++){
        duplicateMovieClip
        ("squares.square
        ", "square"+counter,
        counter);
        duplicateMovieClip
        ("pieces.piece", "piece"
        +counter, counter);
        var temp =
        eval("squares.
        square"+counter);
        var tempimg =
        eval("pieces.
        piece"+counter);
```

3 Lines 11–24: We place our two current MovieClips in position, and attach the Pics clip from the library. We then tell Pics to go to the frame holding the correct image piece. Since we are assembling the grid in the correct order, we can use the counter variable to specify each of the frame numbers. We then assign an onRelease handler which fires the findImgPieces and scanGrid functions.

```
temp._y = tempimg._y
= yval;
temp._x = tempimg._x
= xval;
```

```
temp.itr = counter;
tempimg.itr = counter;
tempimg.attachMovie("
pics", "pics", counter);
var pic =
eval(tempimg+"pics");
pic.gotoAndStop
(counter+1);
temp.onRelease =
function() {
    findImgPieces(this);
    empty._visible = 1;
    empty = this;
    empty._visible = 0;
    scanGrid();
}
```

4 Lines 25–38: Next, we disable the current clip; we'll enable the correct clips when we decide which square has been selected. See how we increment the xval variable inside the rows loop, and the yval variable inside the columns loop. Once finished the loop, we remove the original Square and Piece MovieClips. We then pick the last clips to be placed, and designate them as being empty. Finally, we call the scanGrid function.

```
temp.enabled = false;
grid[rowItr][colItr]
= temp;
xval += 70;
counter++;
}
xval = 0; yval += 70;
squares.square.
killMovieClip();
pieces.piece.killMovieClip();
_global.empty =
eval("squares.
square"+(counter-1));
_global.imgempty =
eval("pieces.piece"+
(counter-1));
empty._visible = 0;
imgempty._visible = 0;
scanGrid();
}
```

5 Lines 39–54: These two functions, enableGrid and disableGrid do exactly as their names suggest. The latter simply loops through the grid's pieces and disables them all. This allows us to prevent the user from interfering with the board before we've processed their interaction.

EnableGrid only enables the pieces within the availablePieces array, preventing the user from trying to move pieces they shouldn't be able to.

```
function enableGrid() {
    for(var i=0;
    i<availablePieces.length;
    i++){
        availablePieces[i].
        enabled = true;
        availablePieces[i]
        ._alpha = 0;
    }
}
```

```
function disableGrid() {
    var counter = 0;
    for(var rowItr = 0;
    rowItr < rows.length;
    rowItr++){
        for(var colItr = 0;
        colItr < columns.length;
        colItr++){
            var temp =
            eval("squares.
            square"+counter);
            temp._alpha = 0;
            temp.enabled = false;
            counter++;
        }
    }
}
```

6 Lines 55–60: FindImagePieces is a key function. When the user clicks a square, we can't move the square itself, because we'd lose track of what squares were where. Instead, we move the corresponding Piece MovieClip. This Piece is the clip that sits in the same position as the square, and hence it could be any of the 15 Pieces, depending on how far into the game we are. So, to track it down, we need to search through the Pieces to find the one in the same position as the clicked square.

```
function findImgPieces(clip) {
    for(var i=0; i<=counter;
    i++){
        var temp =
        eval("pieces.piece
        " + i);
        if(temp._x == clip.
        _x && temp._y ==
        clip._y) {
            temp.
            easeOutMove(empty.
            _x, empty._y, 20);
        }
    }
}
```


Part 3: The scanGrid function

This loops through the puzzle grid to establish the newly available moves...

Our second chunk of code deals with the scanGrid function, which uses a Switch statement to analyse the current empty space. Once it has found a match for the empty space, it works out which square or space the user should be able to move into, and then pushes the correct options onto a freshly-created availablePieces Array. We used this Array in the enableGrid function earlier on.

Switch statements are oft-forgotten yet useful items – they act as a more efficient replacement for the labyrinthine If-Else statement that the following code snippet (and others like it) would otherwise consist of. They're also easier to read and edit, because they're specifically designed to examine a large number of cases. We've not used it here, but Switch also provides the user with the ability to set a default case, which effectively serves as an Else safety net.

1 Lines 1–2: The scanGrid function disables the grid with disableGrid. It then creates a new array called availablePieces. We'll use this to store the places the user can choose. By creating this array each time, we ensure that the previous one is deleted.

```
function scanGrid() {
    disableGrid(); _global.
    availablePieces =
    new Array();
```

2 Lines 3–4: We're going to use a switch statement to establish which pieces we need. The syntax is simple: declare the case you're testing for, then write the code. Include a break statement after each case – this optimises the code by preventing it testing for cases when it already has the answer.

```
switch (empty) {
    case squares.square0 :
        availablePieces.
        push(squares.square1);
        availablePieces.
        push(squares.square4)
        ; break;
```

3 Lines 5–16: We're creating a snippet of code to deal with each of the 16 places the user can click. Ideally, you wouldn't want your switch statement to get any bigger than this – it's reasonably clunky as it is. Note the syntax used to add clips to the availablePieces array: the push() method provides us with a simple way of adding items without worrying about how to extend the Array's length.

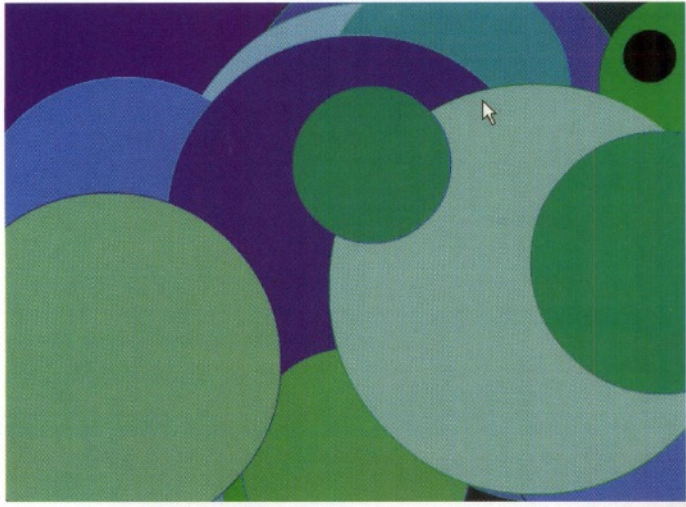
```
case squares.square1 :
    availablePieces.
    push(squares.square0);
    availablePieces.
    push(squares.square2);
    availablePieces.
    push(squares.square5);
    break;
case squares.square2 :
    availablePieces.
    push(squares.square1);
    availablePieces.
    push(squares.square6);
    availablePieces.
    push(squares.square3);
    break;
case squares.square3 :
    availablePieces.
    push(squares.square2);
```

```
availablePieces.
    push(squares.square7);
    break;
case squares.square4 :
    availablePieces.
    push(squares.square0);
    availablePieces.
    push(squares.square5);
    availablePieces.
    push(squares.square8);
    break;
```

4 Lines 17–56: Here we're hard-coding the available squares. In the case of square15, the only places the user can move to are square11 and square14. Once the statement is finished, we call the enableGrid function and are ready for user's next move.

```
case squares.square5 :
    availablePieces.
    push(squares.square1);
    availablePieces.
    push(squares.square4);
    availablePieces.
    push(squares.square6);
    availablePieces.
    push(squares.square9);
    break;
case squares.square6 :
    availablePieces.
    push(squares.square2);
    availablePieces.
    push(squares.square5);
    availablePieces.
    push(squares.square7);
    availablePieces.
    push(squares.square10);
    break;
case squares.square7 :
    availablePieces.
    push(squares.square3);
    availablePieces.
    push(squares.square6);
    availablePieces.
    push(squares.square11);
    break;
case squares.square8 :
    availablePieces.
    push(squares.square4);
    availablePieces.
    push(squares.square9);
    availablePieces.
    push(squares.square12);
    break;
case squares.square9 :
```

```
    push(squares.square5);
    availablePieces.
    push(squares.square8);
    availablePieces.
    push(squares.square10);
    availablePieces.
    push(squares.square13);
    break;
case squares.square10 :
    availablePieces.
    push(squares.square6);
    availablePieces.
    push(squares.square9);
    availablePieces.
    push(squares.square11);
    availablePieces.
    push(squares.square14);
    break;
case squares.square11 :
    availablePieces.
    push(squares.square7);
    availablePieces.
    push(squares.square10);
    availablePieces.
    push(squares.square15);
    break;
case squares.square12 :
    availablePieces.
    push(squares.square8);
    availablePieces.
    push(squares.square13);
    break;
case squares.square13 :
    availablePieces.
    push(squares.square9);
    availablePieces.
    push(squares.square12);
    availablePieces.
    push(squares.square14);
    break;
case squares.square14 :
    availablePieces.
    push(squares.square10);
    availablePieces.
    push(squares.square13);
    availablePieces.
    push(squares.square15);
    break;
case squares.square15 :
    availablePieces.
    push(squares.square11);
    availablePieces.
    push(squares.square14);
    break;
}
enableGrid();
}
```

Welcome to PinkPinkPink

Our collection consists of practical and luxurious lingerie, as well as sports wear. Available in cup sizes AA to J. Please feel free to browse.

Please select brand to view product ranges:

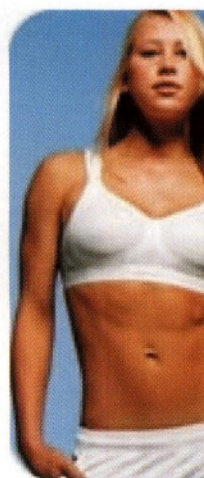
Gossard >>

Panache >>

inkPinkPink.co.uk >

a haven for stylish contemporary lingerie... [search](#)

[CONTACT US](#) >> [SIZE GUIDE](#) >> [ORDER BY POST](#) >> [SHOW](#)



4

Web design tips and tricks

Successful finishing touches require attention to detail. Discover the secrets of the web professionals as they expose their favourite techniques and shortcuts...

174	Return of the code
178	Optimise images for the web
184	50 ways to better web graphics
188	Graphics – Fireworks
190	Tricks of the Trade part 1
194	Tricks of the Trade part 2
198	Tricks of the Trade part 3
201	Tricks of the Trade part 4
203	60 ActionScript tips



Return of the code

Designers with short memories may think that coding is a modern trend, but the web has been there, done that. In days of yore, it was code or die...

Ten years ago, the most complex piece of coding you were likely to see on any web page was the BGCOLOR tag. Before frames, before tables, before PHP and JavaScript and all the rest, clicking a link was about as sophisticated as it got — unless you were a real tech-head, struggling with Perl behind the scenes.

And yet those early pages and sites were indeed coded — entirely by hand. When HTML alone ruled the roost, there was really no other way to create than to dive in, fire up a text editor, and monkey about with tags until you got it right. While it might seem a far cry from today's world of database-driven

sites, the pioneers of the web were certainly coders first and designers last, if at all.

The origins of the web itself are now widely familiar to most designers — particularly its birth in, of all places, a particle physics centre. While working at the Conseil Européen pour la Recherche Nucleaire (CERN) in 1989, Tim Berners-Lee wrote a modest little proposal for a system of information management via hypertext, which was a hot concept at the time.

Working on a NeXTStep computer, Berners-Lee soon created the first browser, which he dubbed WorldWideWeb, also defining vital concepts such as the www.xxx.com URL format, HTTP and HTML along the

way. Although all these technologies were to prove revolutionary, it was the latter which truly opened up the nascent web as a medium for anyone, not just programmers.

Language skills

HyperText Mark-up Language was based on an earlier language, SGML (Standard Generalised Mark-Up Language), which is an international standard for describing text in basic structural units. HTML was designed to be browser-independent; so you could create a web page that would, in theory, look the same on any browser on any platform.

The main attraction of HTML was its simplicity. It resembled a real written

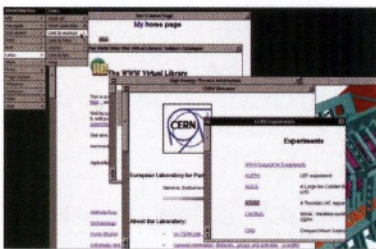




language, with basic rules, punctuation and elements which could be learned in a remarkably short time. You also didn't need any specialist development software to create it — a text editor would do just fine.

In the first year of its life, the web's audience consisted almost completely of academics, students or programmers; access was still restricted, and just configuring an internet connection was a complex business. Thus the first web pages to appear were sober, haphazard affairs, created by people to whom information was far more important than design.

Not that there was all that much you could do with design in early HTML, anyway. The first wave of browsers had no support for



One of the first web pages from Tim Berners-Lee's original browser, running on a NeXTStep computer in 1993. Inline images had just been introduced



Microsoft's FrontPage failed to dominate the web design market, although the program is still being developed today (2003 version pictured)

images at all; in fact, it wasn't until Mosaic 1.0 came along in 1993 that inline images were supported.

Soon, HTML and browsers were being developed at a phenomenal rate, while both Netscape and Microsoft were starting to add their own, non-standard extensions to the



Many pro web developers use the Zend Development Environment to create the backbone of sites in PHP

language. Slinky new tricks such as blinking text and, later, frames became easy attention-grabbers, though the results were horrible.

In short, HTML was a mess. New tags and proposals were being added all the time, and with most people still creating sites by hand or with simple word-processing-like utilities such as HomeSite, you'd be hard-pushed to find two pages that looked similar even on the same site.

The holy grail, of course, was a WYSIWYG web page editor and designer. While there were plenty of editors available, none could enable you to accurately make changes to a page using purely visual tools.

Microsoft had a pop with FrontPage in 1996 (which was actually purchased wholesale from another company). It was a start: you could use familiar tools to design a page, add graphics, and preview the results in your (Microsoft) browser, all with much more ease than hand-coding.

However, FrontPage was incredibly clunky. For most of its advanced effects, it relied on proprietary extensions being installed on the web server, and the code it produced was bloated and horrendously complex. It was fine for home users, but hardly a killer application.

Dream software

That defining program came in the shape of Dreamweaver. Banking on the potential of the internet, Macromedia shifted its focus from CD-ROM multimedia development to web publishing, and in late 1997 it launched Dreamweaver 1.0, based on discussions that the company had had with various leading web designers of the day.

The success of Dreamweaver surprised even Macromedia, with 60,000 people

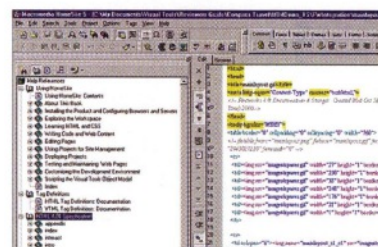
downloading it in the first couple of weeks, and its popularity only grew from there. Before long, both experienced web designers and novices alike were revelling in the ability to produce truly good-looking websites in a fraction of the time it would have previously taken them — if they'd bothered at all.

As competitors such as GoLive joined it, and companion packages such as Fireworks popped up, the web became a riot of colour, sound and design — not all of it good — and a

"The pioneers of the web were certainly coders first and designers last, if at all"

playground for Flash, ActionScript, JavaScript, VBScript and a zillion other scripting tools. Finally, it seemed, the web had been opened up to all and sundry in terms of design.

But as the web matured and technology along with it, a problem with this purely



Now hand-coding is back in vogue, apps such as the long-established HomeSite are making a return

visual approach emerged — one which a few early web designers had always cautioned against. Nick Nettleton of Plum Digital Media soon ran into this brick wall.

"I started designing using GoLive," he says. "I'd design everything in Fireworks, and then cut the whole thing up into images, and put it into GoLive. That's a real designer's way of doing it, and even now a lot of them do a similar thing in Photoshop and ImageReady or whatever, then use Dreamweaver to tinker with it."





“No one could have predicted that getting your hands dirty with code would be the norm in 2004”

Get with the program

Programming languages for the web come in many flavours, each intended for different purposes. Here's a look at the most widely used...

Perl

Developer: Larry Wall

Year: 1987

One of the first cross-platform programming languages, Perl (a made-up word rather than an acronym) has been described as “the duct tape of the internet” and is probably the most popular language in use. It can function as a back-end tool for text manipulation, e-commerce, web servers, and just about anything else you can think of.

HTML

Developer: Tim Berners-Lee

Year: 1990

While no longer really regarded as code per se, HyperText Mark-up Language was (and in some case still is) the bedrock of a huge proportion of websites.

PHP

Developer: Rasmus Lerdorf

Year: 1995

The massively successful server-side scripting language whose main purpose is to generate HTML content. It was created as a set of simple Perl scripts to keep track of accesses to Lerdorf's online

CV, and given the unassuming name of Personal Home Page Tools. PHP 3.0, a complete rewrite created in 1997 by Andi Gutmans and Zeev Suraski, set the tone for its current incarnation. Since PHP is free, easy to learn, and runs on just about any platform, its use has mushroomed.

Java

Developers: Patrick Naughton, Mike Sheridan, James Gosling
Year: 1995

This was created for Sun as an independent, cross-platform programming language. Java is interpreted and runs in its own Virtual Machine within a browser, a feature that has been both its main USP and one of its biggest problems.

JavaScript

Developer: Brendan Eich

Year: 1995

Created for Netscape and originally dubbed LiveScript, the idea was to provide a simpler way to access Navigator's new Java support for non-programmers, although the two are totally different languages. JavaScript got off to a rocky start, but it soon overtook the much-championed Java in popularity.



As with FrontPage, Microsoft missed the boat with its first version of Internet Explorer, which was poor compared to Netscape's Navigator

HTML, though, was never designed as a layout tool, in the way that a desktop publishing package would be. With limited options, designers began using tables to control the layout of a page, merging cells as necessary, sometimes in massively complex nests.

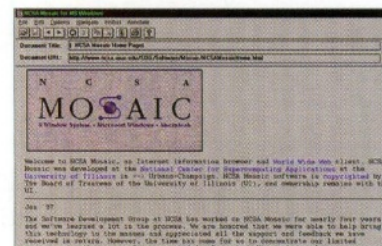
“The problem is that suddenly, the information and the way it's stored can't be used in any other way except through a web browser,” explains Nick. “All those tables don't make sense; you lose the tags and headings which enable different



Created by Plum Digital Media and Scott Bedford, the PinkPinkPink e-commerce site runs largely on PHP and proprietary technologies

technologies to find out about that information. For example, with a properly formatted bit of HTML, Google can scan it and say hey, that's a heading, it's a really important bit of text.

“A lot of web designers now make the mistake of thinking that because everyone has better technology, we've all got big screens,” continues Nick. “But the opposite is true; with more diverse technologies, designers have to think about how things are going to look on a PDA or a mobile phone or whatever. The strength of the internet is its portability; that's the whole point of having information in a digital form — it can be shared.”



Mosaic, developed at the NCSA, was the first browser made widely available, and many people's introduction to the web

And there was still the problem of bloated code. Dreamweaver, while significantly more efficient than early efforts such as FrontPage, might still produce a page of 100K when hand-coding would reduce it to 10K.

“Some designers think it doesn't matter now that broadband is more common,” says Nick, “but you're going to start worrying when the bandwidth on your server gets really high — that's way more expensive than anything else.”

Hand-coding revival

For all these reasons and more, hand-coding once again came to the forefront, at least in the realm of professional



These days, Flash is capable of far more than just annoying little cartoons, being used to create entire front ends or websites

developers. At the very least, stylesheets and CSS are now almost mandatory for all but the smallest site, while those building sites generally turn to PHP, ActionScript or similar programming languages to maintain the infrastructure.

And for people such as Nick, plain old coding tools such as HomeSite are the best

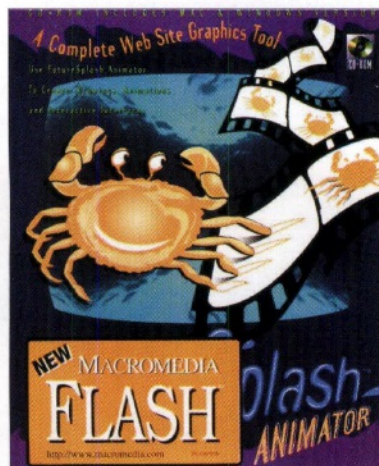


Yahoo from 1996. Although the latest incarnation is far busier, the basic design remains the same – simply because it works

way to do that. "In my opinion, this has been the mistake that Dreamweaver and Flash: they try to be all things to all people," he says. "So we use HomeSite for doing the HTML, and PHP via the Zend Development Environment for the back-end stuff."

Dreamweaver and other visual design tools haven't entirely fallen from favour though. Ryan Carson, co-founder of BD4D and head of Carson Systems, couldn't do without it. "When it comes to being able to combine visual stuff such as graphics with PHP, I think Dreamweaver is really good," he says. "Frustratingly enough, no other tool has the basic functionality of splitting the screen into code and design view, not even Zend. I use it all the time."

Being able to code, he believes, is now a vital requirement for any self-respecting web designer. "I think a lot of designers now are quite afraid when they go to an interview that



FutureSplash Animator, just one of the many incarnations Flash has had en route to its current domination of web animation



Amazon from 1996 – largely a collection of simple HTML links. Conversely, the modern store is entirely code-driven for maximum flexibility

they'll be asked if they can do code too. There's a danger of those people going extinct if they can't adapt."

Nick doesn't believe that should be a



Most professional design sites use a mixture of coding technologies. BD4D runs on PHP, MySQL, JavaScript, HTML, Shockwave, Flash and QuickTime

problem. "In fact it's almost more intuitive than using all this tables malarkey. For example, it enables you to write a text-only site without writing two sets of templates, which means you can share the savings with the client."

No one could have predicted that getting your hands dirty with code would be the norm in 2005; but as many code-wary designers have discovered to their relief, that doesn't mean turning into some sort of obsessed programming geek. Learning to code, even at a basic level and in whatever language, has opened up new doors for most web designers, as well as making their jobs easier in the long run. And that unassuming little invention of HTML remains at the heart of it all. ■

COMPONENT!



No flash in the pan

Flash has come a long way from being a simple doodling tool to a format with a coding language all of its own...

Like many internet coding technologies, Flash has changed considerably over the years. It was originally developed by Jonathan Gay in the early 1990s as Future Splash Animator – a vector illustration tool, but fared badly in a market already dominated by Illustrator and FreeHand.

After a series of changes to its focus – concentrating more on animation, rather than illustration – and its name, Macromedia snapped up the technology in 1996 and renamed it Flash. Initial versions concentrated on the drawing and animation tools, and early uses were largely confined to animated intros or cartoons.

But Flash had always contained a scripting language called ActionScript, based largely on JavaScript. As interactivity on the web became increasingly important, the company developed this side of the product. Thus, by the time Flash MX was released, ActionScript had become a powerful language in its own right, enabling Flash to create complex GUI elements for entire websites.

Flash and ActionScript attract many coders because they combine much of the power of the far more complex (not to mention expensive) Director with rich multimedia graphic tools. Indeed, it seems likely that Director will gradually be phased out in favour of Flash gaining ever more sophisticated coding capabilities.



FILE SIZE
23 KBs

FILE SIZE
3985 KBs

Don't know your JPEG from your GIF, or your lossy from your lossless? Take a look at our guide to getting the best from your web images...

Optimising images



The files needed to complete this tutorial can be found on the cover disc.



Expertise provided by Craig Grannell. Craig is a writer, designer and musician who's been working with the web for nearly ten years. Contact him via his web site at www.snubcommunications.com

FILE SIZE
234567 KBS

for the web

Much online content includes images, but with many users still on dial-up, it pays to optimise your web images thoroughly. If you don't, slow downloads may make users go elsewhere; and even if a user is on broadband, it's still good to ensure your sites download as rapidly as possible.

Compressing and optimising images for the web used to be something of a hit-and-miss affair. All that has changed over the past few years, with both Adobe Photoshop and Macromedia Fireworks including numerous tools aimed specifically at web output. Although in this tutorial we're working with Photoshop, the vast majority of the advice is also suitable for Fireworks users (and, for that matter, anyone using pretty much any graphics editing package with web output features).

Along with explaining how and why to use the common web graphics formats — JPEG, GIF, and PNG — we'll provide some brief guides regarding how best to create online text, optimise and retouch photographs, and output thumbnail images. We're assuming that you're proficient in Photoshop for this tutorial and know how to do things such as work with palettes, create layers and flatten images. ➤



Part 1: GIFs and transparent GIFs

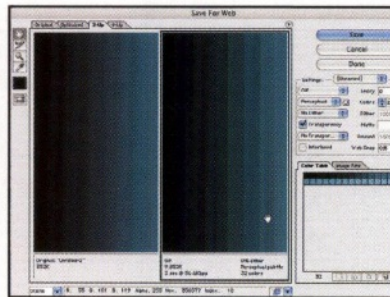
Learn about where, when and why to use the GIF file format...

Don't overwrite original files

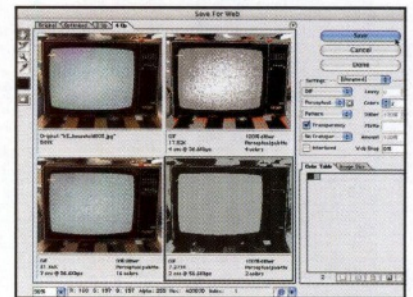
It's all too easy when working on compressing web images to overwrite the original files. But many images — both photographs and designs — can be once-in-a-lifetime efforts. If you compress your sole copy of your high-resolution, top-quality original and overwrite it then shut all files, you can't go back. Therefore, always work with copies of your original files.



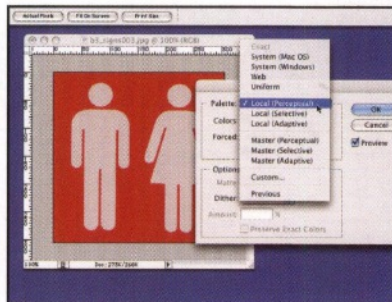
1 The GIF (Graphics Interchange Format) file format is lossless, which means there's no degradation of colour or detail as images are compressed. However, it's restricted to 256 colours, meaning it's best suited to images with large areas of flat and limited colour (logos, text, line art).



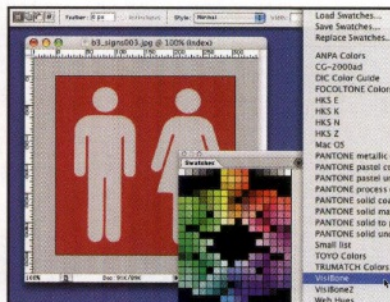
2 The GIF format proves ineffective when used with images that have continuous tones, such as colour photographs. In the example above, you can see how the format's restrictions with regard to the number of colours it supports produces a banding effect, as colours are reduced and grouped to nearest equivalents.



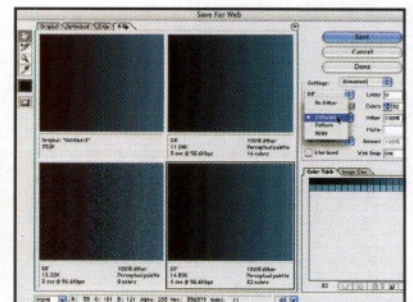
3 Although you shouldn't use GIF to display photographs online, the format's limitations can prove useful for creating special effects. In Photoshop, images are converted to indexed colour, ready for GIF output via Image > Mode > Indexed colour. Reducing the number of colours (say, to single figures) produces posterised and lo-fi effects.



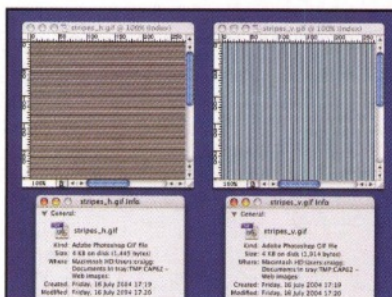
4 Although GIFs are restricted to 256 colours, each image may have a different palette. When converting to indexed colour (and also in the Save for Web dialogue), you can choose a palette, such as Perceptual (prioritising colours sensitive to the human eye), Selective (mixing those colours with web-safe colours) and Web (see next step). Perceptual is generally a good choice.



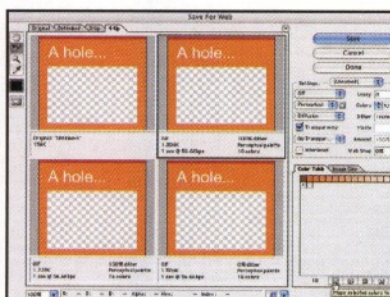
5 The Web-safe palette was intended to cater for monitors restricted to 256 colours. Because Windows and the Mac had exclusive system colours, only 216 were deemed web-safe, and these comprise the Web palette. With modern monitors not suffering this restriction, the Web palette is generally considered archaic, unless authoring specifically for legacy kit.



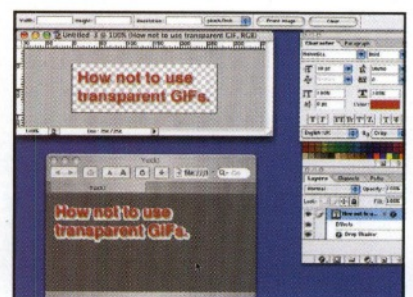
6 With GIF mostly being used for flat-coloured images, dithering is usually best turned off. If part of the image has continuous tones, dithering replaces the banding seen in Step 2 with a pattern to simulate the colour gradients. Diffusion applies random patterns, Pattern applies a half-tone pattern, and Noise works like Diffusion but without diffusing the pattern across adjacent pixels.



7 Because of the way GIF compression works, it pays to have continuous areas of horizontal colour. Try exporting striped images as a GIF, such as the ones shown above — the image with horizontal stripes will have a smaller file size than the one with vertical stripes.



8 The GIF89 file format is commonly referred to as a 'transparent GIF', but you actually remove selected colours so the background shows through. Transparent areas of a PSD file automatically carry over to the Save for Web dialogue, and more can be selected in the Colour Table tab and made transparent by clicking the icon shown above.



9 Transparent GIFs work best when removing areas bounded by straight lines (squares and rectangles); removing irregular shapes, or anti-aliased areas, causes the effect shown above when the image is placed on a coloured background in a web browser.



Part 2: Working with JPEGs

Find out how to compress JPEGs and see what images the format is suited to...

Scanning and resolution

Never output images for the web at anything other than 72dpi, because that is the standard. However, although web images are at a low resolution, you should still scan things in at a higher resolution, to provide yourself with more pixels to play with for cropping and other editing tasks.



1 JPEG stands for Joint Photographic Experts Group, and is primarily used for images that have continuous colour tones and transitions. JPEG supports millions of colours, and is ideal for the web.



2 JPEG compression is lossy, so as compression increases, information is removed. Never set the Quality level below 40 – set it as high as you can without compromising download times.



3 You can also save images as Progressive JPEGs, which means the image downloads in multiple passes, and immediately presents the user with a low-resolution version before the rest downloads.

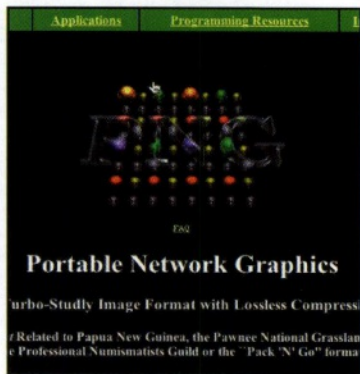
Part 3: Alpha transparency with PNG

The new kid on the block – but how do browsers cope with PNG?

Check your gamma settings

Depending on the operating system you're using, you may have to tweak your colour settings. Typically, displays on Windows tend to be darker than those for the Mac. This means that subtle dark colours created on a Mac may be indistinguishable on the platform that the majority of people use. However, Windows users should also be aware of the reverse: subtle, very light colours may not display well on a Mac. Mac users should check their OS and software settings, and set them to emulate PC gamma when creating web images; and Windows users should, on occasion, do the reverse.

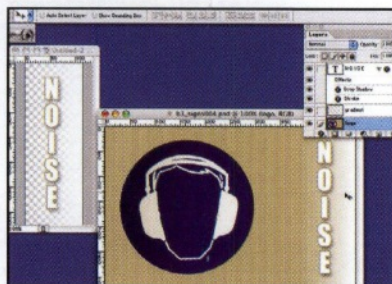
The sound of PNG



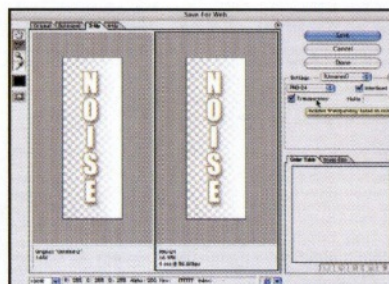
The Portable Network Graphics file format (PNG, charmingly pronounced 'ping') was once hailed as a successor to other web image file formats, and it's easy to see why. In PNG-8 form, it works rather like a patent-free replacement for GIF, as it supports indexed colour. However, in PNG-24 form, things are more interesting. Its support for greyscale and images with millions of colours makes it a challenger to JPEG, but PNG also supports alpha transparency – and unlike GIF89s, this is true alpha transparency.

Unfortunately, despite the massive potential of the format, various browser manufacturers were slow to react, grudgingly supporting some features of the format, but not all of them. Even in these enlightened times, Microsoft's behemoth – Internet Explorer – still doesn't support PNG's alpha transparency. PNG transparency is instead shown as solid white in Microsoft's browser.

Although this rather throws a spanner in the works, there is a very messy workaround, which enables you to use conditional comments and proprietary code to get Internet Explorer to behave (as of version 6, at least). We detail the workaround in this section's walkthrough, although, due to the nature of the code, we suggest you use it sparingly. www.libpng.org/pub/png/



1 PNG can be useful for overlaying graphics, perhaps if a logo is to remain constant but the caption is to change. Here, we've created an overlay with alpha channels (a white foreground to transparent gradient, and a drop shadow on the text).



2 The relevant portions of the image are copied to a new file. Then, in the Save for Web dialogue, choose PNG-24 from the formats dropdown menu, and ensure that Transparency is checked. Photoshop should append the file extension, but if not, use .png.



3 At the top, you can see how the alpha transparency appears when the workaround has been applied (this is how it's displayed in other browsers by default); at the bottom is IE's default effort, with the transparency shown in white.



Part 4: Using images for online type

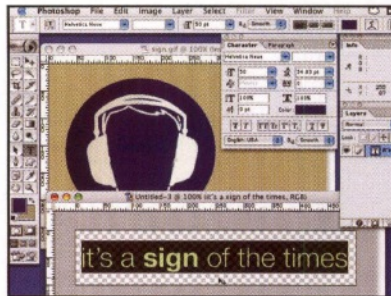
Get around the limitations of web-safe fonts by using images for display copy...

Don't use other formats

Although some browsers tolerate the use of formats other than those discussed in this walkthrough, they generally do so by relying on system software-level that all users may not have installed. Also, the formats we've mentioned are tailored specifically for web output. Therefore, don't be tempted to use TIFFs, BMPs or any file format other than JPEG, GIF or PNG — if you do, file sizes will balloon and not all users will see your work.

Anti-aliasing

When experimenting with anti-aliasing, bear in mind that Windows doesn't anti-alias body copy by default, so you won't be able to match graphical text to HTML text on Mac and PC.



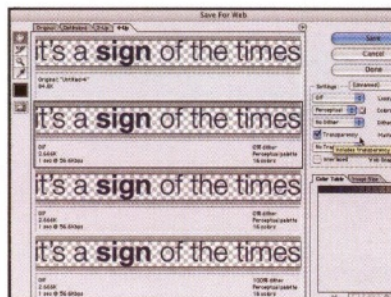
1 There are fewer than half-a-dozen web-safe fonts of any quality, hence many designers like to use images for display copy. The web's resolution is much lower than print, so work with highly readable fonts at a size that's comfortable to read.



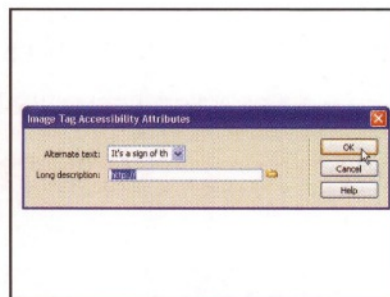
2 Experiment with anti-aliasing to get the most suitable result for your site. Provide contrast between text and background colours, because it's harder to read screen-based text than text in print.



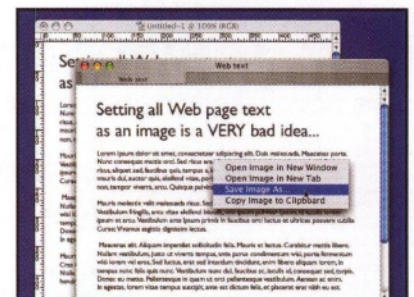
3 Flatten the image on to your background colour, then crop it. Float the layer and use the Magic Wand with Tolerance set to 1, and Contiguous and Anti-alias off, to select/delete the background colour. The web page's background appears in these areas.



4 In the Save for Web dialogue, choose GIF as the file format and ensure that Transparency is checked. Unless you have very flamboyant text, the image should have few colours, resulting in a file with a small download time.



5 When adding the image to your web page, always remember to include alt text, for those people surfing with alternate or assistive devices. By and large, this should replicate the text within the image. However, if the image has a function (such as if it's a link), the alt text should convey that too.



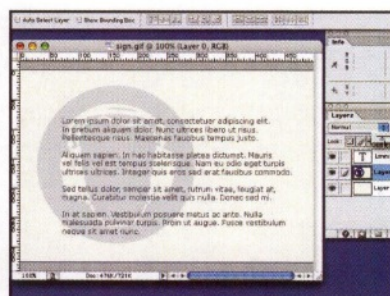
6 You should use images for web-based text sparingly — it can't be read by search engines, can cause accessibility problems, prints badly, and cannot be copied to a text file. Under no circumstances should you use images for body copy — unless you really want to annoy your site's visitors.

Part 5: Effective watermarks for the web

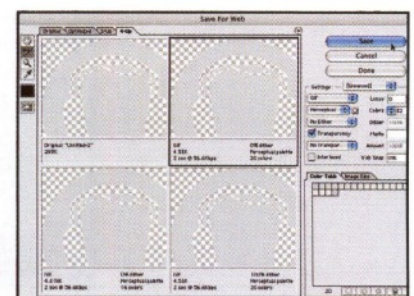
When used sparingly and carefully, background watermarks can be effective online...



1 Subtle photographic watermarks don't work well on the web. Because the low resolution makes it hard to read on-screen text, losing text in murky images makes it even harder. Therefore, restrict background images to simple line art.



2 Ensure that the contrast level between watermark and text is strong (overlay dummy copy to test this), and that the image's contrast with the background colour is fairly slight, so it doesn't distract too much from the page content.



3 Once satisfied, use similar techniques to those used when exporting text: flatten the image, float the layer, and use the Magic Wand to delete the background colour. Then export the image as a GIF in the Save for Web dialogue box.



Part 6: Optimising and retouching photographs

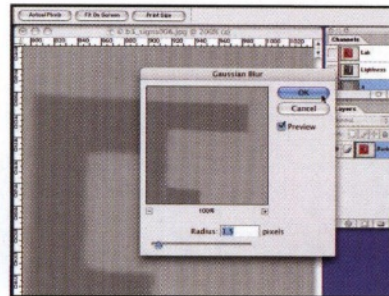
Turn a scanned photograph into an optimised and edited web image...

Wrecking images with watermarks

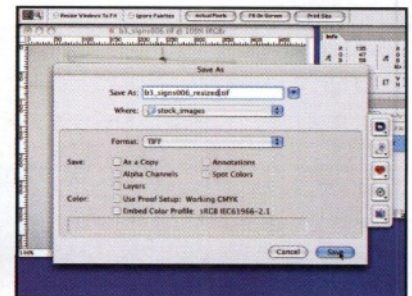
If you're a stock photography house, it may be forgivable to slap watermarks all over your web images. But if not, you'd be mad to do it. Copyright exists on the web, the same as everywhere else, and even if someone is audacious enough to download your website images, a compressed, low-res JPEG is hardly going to be of use to them. For portfolio images that may be sent between agencies, it's sometimes wise to put your name and URL in a corner, in small type, but that's about all.



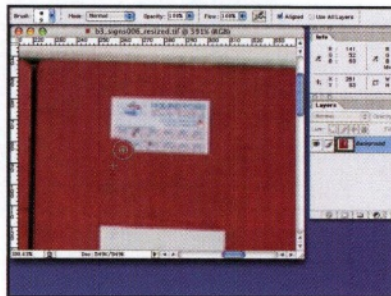
1 The chances are your photographs may lack some punch. Shots from compact and low-end digital cameras, scanned in using low-end scanners, can usually be improved via Image > Adjustments > Auto Colour. If Photoshop overdoes it, go to Edit > Fade Auto Colour to reduce the effect.



2 Scanned images may have moiré, and digital images will almost certainly have JPEG artefacts. These can be reduced by converting the image to Lab colour and running a two-pixel (or thereabouts – it depends on the pixel dimensions of your image) Gaussian Blur on the 'a' and 'b' channels.



3 Convert the image to RGB and, if satisfied with the colour, save the image as an uncompressed TIFF file. Ensure it's flattened, then resize it, and save it as another TIFF. Doing this means that if you need to resize or recompress it later, you can go back to those two uncompressed versions.



4 Print designers may balk at this, but when retouching web images using the Clone tool, it's often best to do so on the low-resolution versions. This is because there are fewer pixels, and you can work much faster, but get satisfactory results for screen.



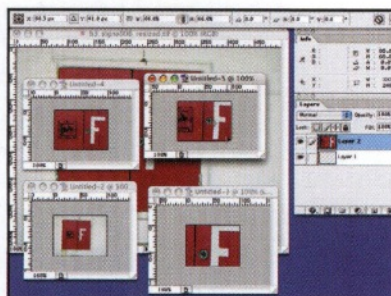
5 When images are reduced in size, they tend to take on a slight blur effect, reducing the impact of line and colour. Bring back the image's punch by using an Unsharp Mask filter, set to approximately 20 per cent.



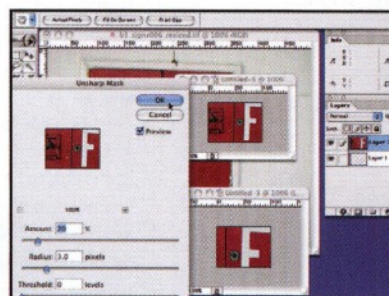
6 Use Save for Web to export the image as a JPEG. Don't over-compress important images: people would rather wait an extra second to see a great image than get a shoddy one. Use the Hand tool to check how your settings affect each part of the image before confirming your choice.

Part 7: Creating thumbnail images

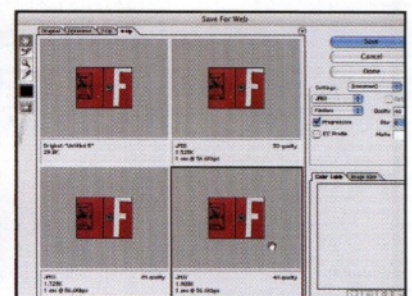
Crop, resize and contrast images, then export them to miniature JPEGs...



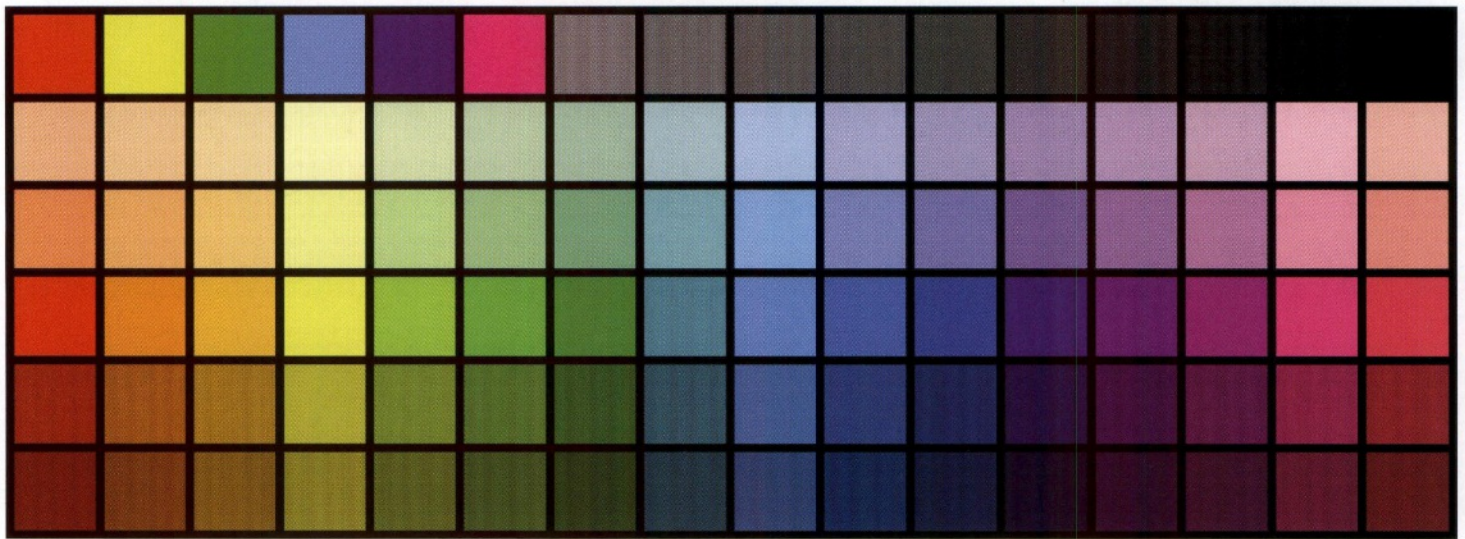
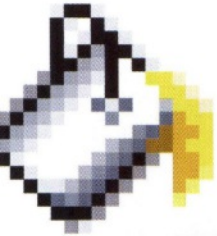
1 Online galleries often use thumbnails for navigation and a visual preview. Depending on your images, you may wish to display resized images or cropped portions. In either case, ensure it's clear what the images are.



2 Prior to resizing the image, flatten it to ensure it's not on a floated layer (otherwise Photoshop anti-aliases its edges). Once you've reduced it, use the Unsharp Mask filter to bring back some of the lost detail.



3 Thumbnails are there to provide a small visual preview, rather than to be works of art. Therefore, the quality level isn't of paramount importance. If you have many of them on a page, export them with a low-quality setting (of around 35).



50 Ways To Better Web Graphics

50 indispensable tips, tricks and techniques to help you generate great graphics for your site.
Karl Hodge leads the way...



Those who control the web have largely overlooked web graphics. The World Wide Web consortium has standards for CSS and XHTML but it doesn't have a general standard governing web graphics. The few attempts the W3C has made to create standard image formats are only just beginning to gain ground. The PNG format was a recommendation of the W3C back in 1996, but Internet Explorer still doesn't support it fully. Similarly, the SVG vector standard is only just beginning to find its way into the mainstream.

Instead, standards have evolved through

usage and browser support, meeting the challenges of an ever-changing web. It's over a decade since the release of the first graphical web browsers and a lot of knowledge has been forged since then, much of it through trial and error. Ever up for new and exciting challenges, we're going to try to squeeze all that knowledge into the next four pages. We'll admit to cutting some corners along the way — but what you'll get are 50 nuggets of prime graphics advice. With hidden features explained, myths exposed and tricks uncovered, these tips are designed to make your creation and use of web graphics better all round.



Image formats

WHICH IMAGE FORMAT SHOULD YOU USE FOR WHICH JOB?

GIF Graphics

The GIF (Graphic Interchange Format) has traditionally been the web's default for simple images. Normally found as a 256 colour, fixed palette format with built-in compression, it's ideal for bitmap images with limited colour transitions. Use them for buttons, logos and backgrounds.

JPEG Photos

JPEG's are named after the Joint Photographic Experts Group. With 16.7 million colours available in every file and aggressive compression built-in, they've been adopted as the standard format for displaying photographic imagery on the web.

PNG Pictures

PNG (Portable Network Graphics) images are still outnumbered by older formats on the web. They have several advanced features you won't find in other image formats – like variable alpha transparency – but some browsers don't support all those features. Use them in place of GIFs.

BMP Bitmaps

BMP files are entirely unsuited for use on the web. In comparison to formats such as the JPEG, their compression capabilities are poor, which makes them harder to use successfully. BMP files are not supported in any browser, either, except Internet Explorer.

SVG Illustrations

The W3C endorsed SVG (Scalable Vector Graphics) format is poorly supported so far – though Mozilla was projected to incorporate inline support in the first quarter of this year. If IE follows suit, SVG could bring vector graphics into mainstream page layouts.

Flash Imagery

You can import most standard vector file types into Flash and then save them in the bandwidth-friendly SWF format. With 98 per cent of all browsers already equipped with the Flash plug-in, the SWF format has become a widely accepted, none-standard standard for including vector imagery in web pages.

QUICK TIP IMAGEREADY SWITCH

Switch to ImageReady in Photoshop without clicking the toolbar icon using Ctrl, Shift and M. Use the same shortcut to return to Photoshop.

QUICK TIP PREVIEW PICTURES

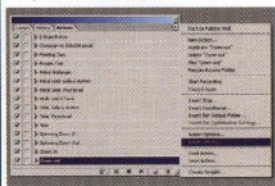
Use Ctrl Y to cycle through view modes in ImageReady. To preview an image directly in your default browser hit Ctrl, Alt and P.

QUICK TIP QUICK OPTIMISE

The function key F10 toggles the Optimise palette on and off in ImageReady.

QUICK TIP BATCH PROCESS

ImageReady has a batch commands just like Photoshop. You'll find them hidden away in the Actions palette.



QUICK TIP SO 20TH CENTURY...

Some web graphic techniques have been done to death. Avoid obvious drop shadows and excessive 3D bevelling on text in particular.



Colour me creative

USING COLOUR ON THE WEB IS FRAUGHT WITH THE LEGACY OF LIMITED PALETTES AND BROWSER QUIRKS...

The 216 Colour Palette

In every web graphics and design package you'll find a special swatch of 216 web safe colours. This palette was, and still is, a set of fixed colours built into browsers for use on 8bit displays. Unless you're specifically designing for 256 colour resolution platforms, like some PDAs, it's largely redundant now.

On The Other Hand...

The web safe palette does make it much easier to match colours in your page to those in a web safe image. In HEX code, colours are written in the syntax #RRGGBB – where R is red, G is green and B is blue. The 216-colour palette is limited to combinations of six values: 00, 33, 66, 99, CC and FF. So #00FF00 is web safe while #CCDD06 is not.

Colour Matching

The limited web palette also makes it much easier

to come up with sympathetic or matching colour schemes for a site. An online tool called Color Schemer uses the 216-colour palette to help you do just that at www.colorschemer.com/online.html.

Web Smart Colours

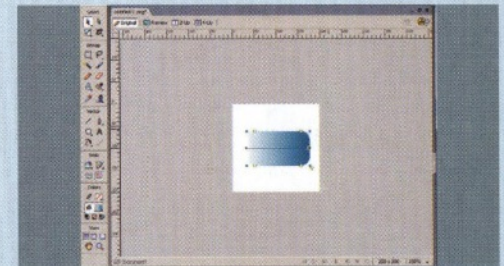
A set of 4096 web smart colours is the more contemporary alternative to the original web safe palette. This uses any three pairs of HEX digits, so #002233 is web smart while #0FG43D is not.

Work Safe or Smart

In Photoshop or ImageReady there are several versions of the web safe palette already built-in. Choose them from the Swatches menu. To make your own web smart colours, go to the Color palette, open the drop-down palette menu and set the sliders to Web Color Sliders, then enter the HEX notation manually in the boxes on the right-hand side.

A SOFT BEVELLED BUTTON

This technique is less harsh than Fireworks' built-in bevel effect, and can be applied to Photoshop or other tools



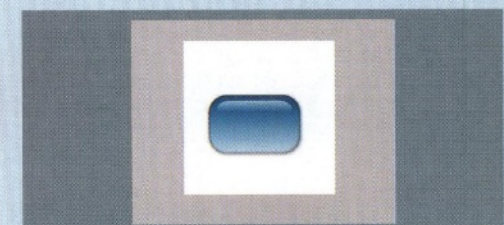
Step 1

Create a new 200x200 pixel document in Fireworks and draw a rounded rectangle in the middle. Select the rectangle and in the Properties panel choose Linear as the fill type. Click on the Fill Colour swatch in the Properties panel and click the black colour slider. Enter #006699 into the text box.



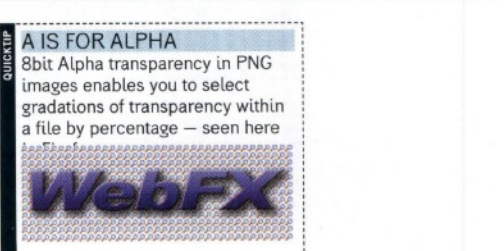
Step 2

Rotate the gradient so blue colour is at the top. Go to the Effects section in the Properties panel and choose Shadow → Inner Shadow, reducing the distance to zero pixels and the opacity to 65%. Next add a Drop Shadow, reducing the distance to two and opacity to 50%.



Step 3

To make this bevelled button into a glass button, draw a small rounded rectangle along the top and fill it with a white and black linear gradient. Go to the Layers palette and set the Blend Mode to Screen. Rotate the gradient so the lighter part is at the top.



QUICK TIP A IS FOR ALPHA

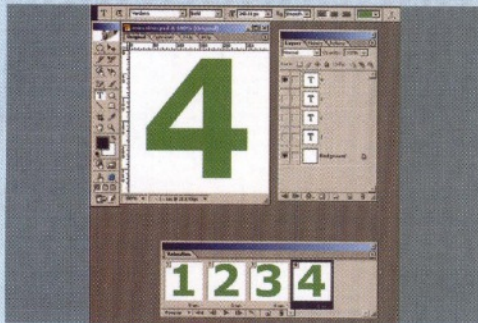
8bit Alpha transparency in PNG images enables you to select gradations of transparency within a file by percentage – seen here





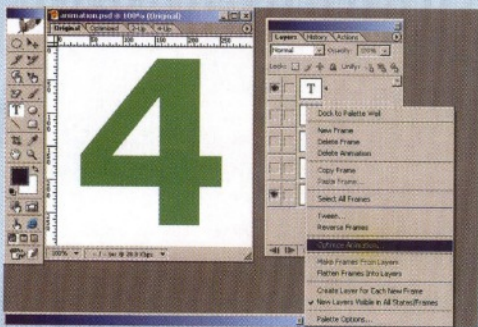
OPTIMISED GIF ANIMATION

Follow these simple steps to streamline your animated GIF files with ImageReady's simple optimisation procedure



Step 1

Animation is created easily in ImageReady. You simply create a new Layer for each frame of animation, then in the Animation palette click the Duplicate current frame icon to make as many frames as you need. Set the visibility in the layer palette to correspond with each frame of animation.



Step 2

To optimise the output, open the Animation palette menu and choose Optimize Animation. Accept the defaults and click OK. You can optimise GIF animations created in other packages within ImageReady in the same way.

QUICK TIP SHAPELY TEXT

Select all your text elements in Fireworks PNG files and choose Convert to Paths in the text menu before saving. This prevents missing font problems.

QUICK TIP FIREWORKS SAVE

The Quick Export button in the top right corner of each document window gives you a fast way to output images and code to other Macromedia programs.

QUICK TIP EDITABLE FILE

Save an editable backup of your source files at the same time as you export your web optimised version.

Photoshop and ImageReady actions

THESE BUILT-IN TIME-SAVERS ARE GREAT FOR SPEEDING UP WEB GRAPHIC PRODUCTION

Quick GIF

Hidden in Photoshop's Actions palette is a one-click tool to convert images to web-ready GIFs. Select Production from the palette menu, then scroll down through the list until you find 'Save As GIF89a Web Palette'. Press the 'Play' button to apply the Action to the current image.

Instant JPEG

In the same ACTIONS category, you'll also find JPEG Medium and Make Button Actions. Any of these actions can be incorporated into an automated Batch process – enabling you to apply an action to a folder full of files. Go to File → Automate → Batch.

Dynamic Resize

Create an action that resizes a batch of images to roughly the same dimensions, using the Create New Action icon in either the Photoshop or ImageReady

Actions Palette. Give it the name Resize Dynamically and press record. Go to Image → Image Size and edit the Width or Height of the image only. Press Stop to finish.

Action Stations

Photoshop or ImageReady actions can be assigned keyboard shortcuts – so you can convert an image into a web ready JPEG or GIF with a couple of key-presses. Select the action you want to assign the shortcut to and open the palette menu. Choose Action Options and select a Function Key combination from the drop down box.

Little Droplets

Actions can be incorporated into Droplets – scripts that you can drag image files to. The host application, either Photoshop or ImageReady, launches and processes the file. Go to File → Automate → Create Droplet to launch the droplet dialog.

Image slicing

CREATE FULL WEB PAGE DESIGNS IN YOUR FAVOURITE IMAGE-EDITING PACKAGE? IT CAN BE DONE...

Image What?

Creating a web layout in an image editor, then cutting up the pieces and arranging them using CSS positioning or tables is called image slicing. Using a consistent grid-like structure in your page designs will make it much easier to stitch your design back together in a web authoring tool like Dreamweaver.

Trace Elements

Most modern web authoring packages allow you to insert a tracing image into your document. Save a copy of your bitmap page design before you slice it up and use that as the tracing image. Assemble your page over the image using CSS Layers.

Consistent Colours

Try to use the same colour palette while you are optimising and saving individual slices – otherwise you could end up with subtle (and not-so subtle) colour discrepancies throughout your design. Another reason to stick with web smart colours.

Table Alignment

Once you have your image pieces prepared, take care when placing the images into table cells. It's best to use the same cell alignment parameters for every slice. If you don't, you could end up with gaps in the image.

Empty Cells

There should be no empty table cells in your page design as they can collapse in certain browsers. One way to prevent this is to create an invisible, single-pixel GIF image in your bitmap editing package and insert an instance of that in each empty table cell. Ensure that you resize the image using the height and width parameters of the tag to fit the size of the cell.

Flat Colour

If you use web smart colours to create your original design you can easily replace an area of flat colour in your design with a table cell or CSS layer containing an invisible image as described above. Use the background attribute of the cell or layer instead.

QUICK TIP INSTANT ACTION

Create Actions quickly by dragging a command from the History palette, hitting F9 to activate the Actions palette and dropping them there.

QUICK TIP IE HACK

Internet Explorer doesn't display PNG transparency correctly, but it is packaged with an alpha filter... You'll find a script that fixes IE's oversight using this at snipurl.com/13s9.

QUICK TIP GIF VS. PNG

PNG compresses in the vertical plane as well as the horizontal one, resulting in smaller file sizes for 8bit images when compared to GIFs.



QUICK TIP LOSSY JPEGs

JPEG's compression algorithm is lossy, meaning you get less detail the more compression you use. Below a ratio of 60:1 JPEG compression – like the image on the left – looks pretty lousy.





Tools you can use

PHOTOSHOP AND FIREWORKS AREN'T THE ONLY IMAGE EDITING PACKAGES ON THE BLOCK

Xara X

Xara X remains the most underrated of web graphics production tools. It costs a measly \$179 (about £90) and has more extensive vector graphic tools than Fireworks or ImageReady, combined with similar web graphics features.

www.xara.com

Paint Shop Pro 9

JASC's venerable Photoshop-a-like began life as a shareware tool aimed at creators of bitmap graphics on the web. It had optimisation and button making features before any other package – and still retails for a reasonable \$95 (£44) as a download.

www.jasc.com

The GIMP

An open source Photoshop clone that debuted on LINUX, there are now Mac OS X and Windows versions

of The GIMP. The interface is tricky to get used to, but there are lots of bundled web graphics features available in this powerful tool.

www.gimp.org

DeBabelizer LE

DeBabelizer is to file conversion what Photoshop is to image editing. This version of the tool, retailing at \$99.95 (£54) has more web graphic conversion features than any mainstream editing program.

www.debabelizer.com

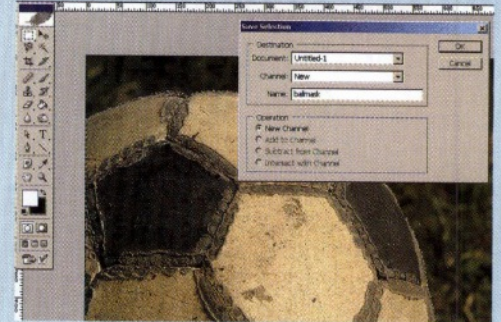
LView Pro

Another tool that's been around for ages, Lview Pro has cribbed a number of Photoshop's more popular web features. It's best known as an image viewer though, with support for many file formats.

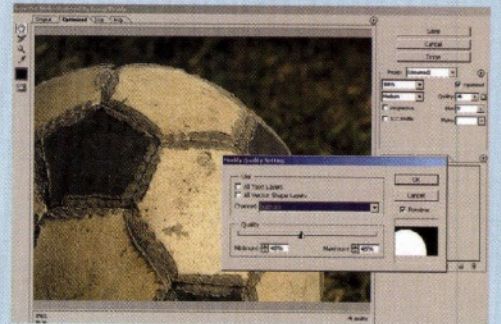
www.lview.com

SELECTIVE IMAGE COMPRESSION

You can control which areas of an image you optimise in Photoshop using masking techniques



Use your favourite selection tool to trace around the focal element in the photograph. When you're happy with your selection, right-click within it and choose Save Selection. Leave the channel set as New and give the selection a name.

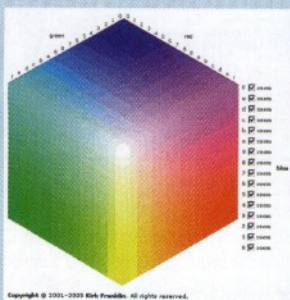


Go to File → Save for Web, then click the 'Use a channel to modify quality setting' icon next to the Quality drop-down menu. Choose the channel you created at stage one from the Modify Quality Setting menu and set the compression quality using the slider. You can preview the effect before saving the image.

COLOUR CONVERSION

If your image editor only has RGB sliders, don't worry – you can still get the 4096 web smart colours using this table. Web smart colours are composed of any three pairs of HEX digits, so the web smart colour #FFAA00 would be rendered in standard RGB notation as 255, 170, 0.

HEX	RGB
00	0
22	34
33	51
44	68
55	85
66	102
77	119
88	136
99	153
AA	170
BB	187
CC	204
DD	221
EE	238
FF	255



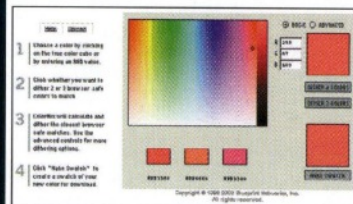
The 4096 colour web smart palette, seen here as a colour cube at www.morecrayons.com

QUICK TIP BROADBAND BITMAPS

Balance image size considerations with image quality. Try to keep total page sizes below 70–80K.

QUICK TIP EXTREME GRAPHICS

Dithering two or three web safe colours together to make a third is quite anachronistic now that most are browsing on monitors displaying millions of colours – even though there are tools available like www.colormix.com.

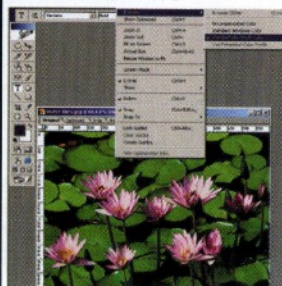


QUICK TIP MAC VS. PC

Windows monitors are darker than Mac monitors – so images created on Windows can look washed out on Macs. Check graphics on both platforms.

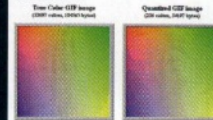
QUICK TIP MORE MAC

Fireworks lets you preview how an image would look on the other platform. Go to View → Macintosh/PC Gamma. In ImageReady go to View → Preview.



QUICK TIP THE GIF MYTH

The GIF file format actually supports true colour – but this behaviour is unsupported in most applications. Visit phil.ipa.org/tc.html for more detail.



QUICK TIP ANIMATED ROLLOVER

Replacing the second image in the rollover with an animated GIF gives a Flash-animation style effect without using Flash...

QUICK TIP ANOTHER GIF MYTH

The Unisys LZW compression patent, requiring that application developers pay licence fees to include GIF support in their programs, expired worldwide in 2004.



Graphics: Bringin' the bling in



All the components needed to complete this tutorial are on your cover disc.

More effects magic to bring impressive effects to your pages with **Scott Bedford's** graphics know-how

Fans of Fireworks often find themselves in the minority, which is odd considering that when it comes to knocking out web graphics, and even whole web sites, Fireworks is definitely one of the easiest and most versatile graphics applications. Of course, any designer worth his salt keeps a copy of Adobe Photoshop to hand, but when it comes to mixing vector graphics with pixel-based imagery Fireworks is just so much easier. If you don't believe it, (or

haven't ever used Fireworks) download the 30-day trial and give this tutorial a go — the proof, in this case, is in the doing. It's a quick and easy exercise that shows off some of Firework's vector drawing capabilities.

Another really cool thing about Fireworks is the way it can turn graphics into symbols. If you've used a graphical device, such as an arrow, repeatedly throughout your design you are able to edit all occurrences of that device by simply editing the master symbol within the library. For designers who really

enjoy tweaking a design, this is a massive time saver.

Editing symbols will be familiar to experienced Flash-users but this tutorial will cover the essentials. We'll be digging up a scrolling heart and adding some bling to the graphics.

You can find a copy of my finished file on the cover disc to use as a reference or, as always, you can visit www.brick-on-the-egg.co.uk/net/webgraphics to view the final results. ■

Part 1: Customised bullet points

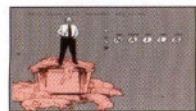
Personalise your page appearance...

RESOURCES ARTISTIC APPEAL

A whole new world of Art

Flash gave artists a new tool and the web gave them a new audience. Check out this artistic collaboration for more inspiration.

www.hoogerbrugge.com



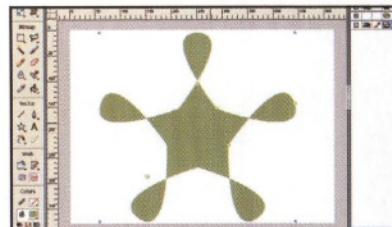
Creative comedy

Like Hoogerbrugge, Mantlepies was created by two bored art directors as an outlet for their comedy ideas.

www.mantlepies.com



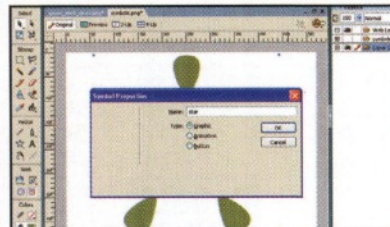
1 Bullet points may perform a useful function but they aren't great to look at. We'll be creating a more interesting replacement (see above example) using Fireworks. Using Fireworks will make this process easier as you can convert graphics into symbols making them really simple to resize and edit later.



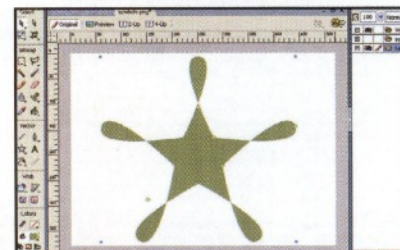
4 One final adjustment and we are done. Click the third handle from the left and drag it a short distance to the right until the points of the star become fatter and more rounded. That's it, one simple way to create a unique designer icon. To finish, we'll quickly save the graphic as a symbol.



2 Firstly, create a new document (with a white background) 500x400 pixels in size. Find the shape tool in the tool panel, by default it's a rectangle, and then click and hold until a list of options appears, select the star option. Make sure the foreground colour isn't white and then click on the canvas.



5 With the star still selected click Modify > Symbol > Convert to Symbol, choose a name and click OK. You will now be able to resize, rotate, skew and duplicate the symbol on the canvas without affecting the master symbol; this is edited by double clicking the symbol or by using the library palette.



3 You should now be looking at a five pointed star with a number of yellow transform handles scattered around it (if your star doesn't have five points, adjust it by dragging the handle on the left). Otherwise, drag the handle on the right until the star develops inverted ends similar to the thumbnail above.



6 This tutorial may seem easy, but that's the beauty of it. By exploiting the drawing tools that come with most graphics programs, you can create icons without being able to draw. As a means of final encouragement the above graphics were created by only using circles and rounded rectangles, go on have a go.

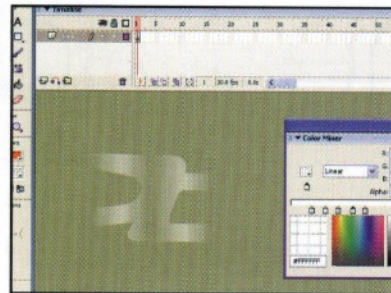


Part 2: Pep up your pages

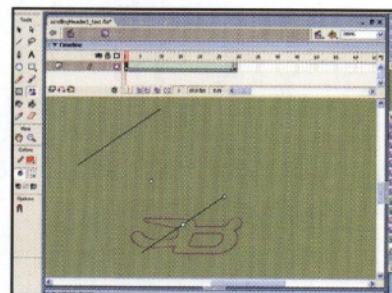
Sparkles, glints and flares...



1 Firstly, open up the Flash scroller on the CD. Then create a movie clip called Bling and place it on a new layer above the layer named logo. Right-click the Bling movie clip and select Edit in Place, use the Pen tool to trace the outline of the logo.



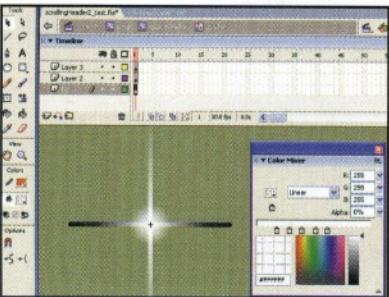
2 With the shape deselected, click and drag on the outline to fine tune it. Make the shape white and then delete the outline. With the shape still selected use the Colour Mixer to create a linear fill consisting of a transparent (alpha 0%) background with two soft bands of white running across it.



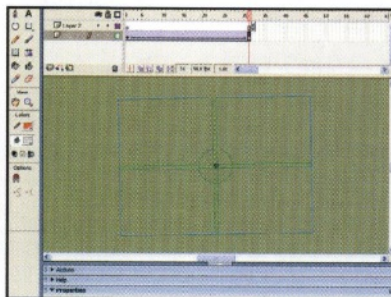
3 Select the Fill Transform Tool and click on the shape. Use the transform handles to rotate and move the fill up and to the left. Add more frames to the time line and set up a shape tween. Select the end key frame and, using the Fill Transform Tool, move the fill down and to the right.

Rise to the challenge

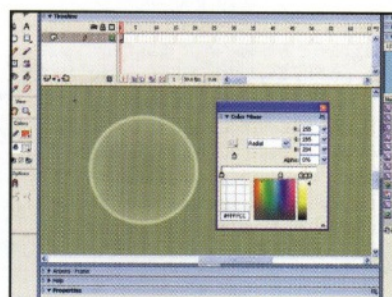
Don't just use popular devices such as indents, bullet points and arrows because that's what everyone else uses. Challenge everything. Try creating emphasis and impact by playing with composition or by creating your own unique graphics.



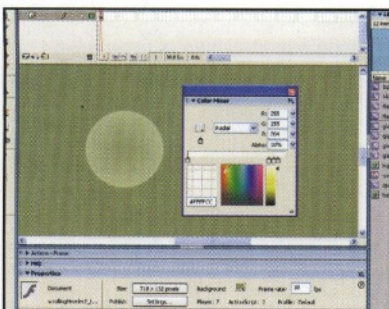
4 Publish the movie. All being well, two bands of light should flash across the logo. Now make a new movie clip called Sparkle, in it create a cross and fill it with a linear fill that fades to transparent. Create a circle in the middle and fill it with a radial fill that also fades to transparent.



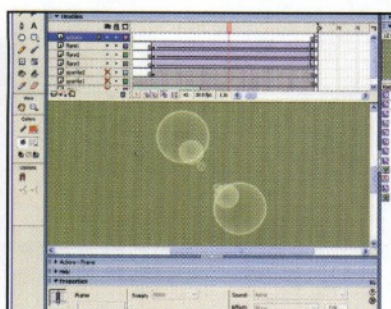
5 Move the Sparkle clip so it sits over the logo. With it still selected, convert it into a new movie clip called SparkleHolder, open up this clip and create a motion tween using the Sparkle clip inside. On the last key frame, rotate the sparkle clip by 90 degrees and turn the alpha to 0%.



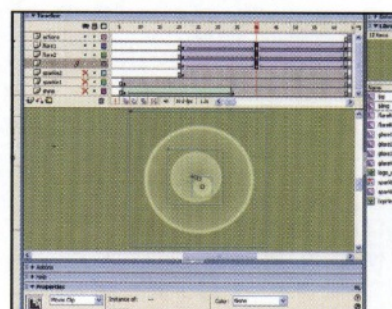
6 Publish the movie again. A sparkle should appear, start rotating and then fade away. Now for the lens flare effect. Create a new movie clip called flareRing1, inside draw a circle and use the colour mixer to add a radial fill that looks a little like a halo, see the thumbnail above.



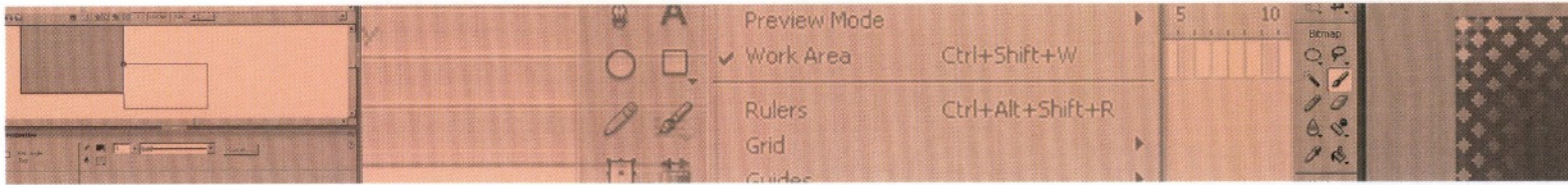
7 Once again, create another movie clip and name it flareRing2. Inside, draw another circle and use the Colour Mixer to fill it with a radial fill that consists of a partly transparent colour that fades to transparent at the centre. Open up the Bling clip again and create three new layers.



8 Drag flareRing1 onto one layer and flareRing2 onto the other two. Resize and move the clips so they match the top left of the thumbnail (thumbnail shows both the start and end positions). Finally, create three motion tweens and move the clips down and to the right until they match the bottom right of the thumbnail.



9 At the centre of each tween add a key frame and then reduce the alpha transparency of the first and end key frames. That's it. Publish one last time to check out the full, bling effect. If you want to take a closer look at this process, check out the finished FLA file on the cover disc.



Tricks of the trade

Web specialist **Ryan Carson** reveals his handy web building tricks that you can use to improve your designs...

Making CSS Work on Netscape 4

Netscape 4 won't use CSS at all, unless you tweak the browser settings. Open up the preferences dialog and check the boxes for both style sheets and JavaScript.

GIFs vs JPEGs

Save an image as a GIF if it has flat colours or needs transparency. If your image is a photo, save it as a JPEG.

In this first part of Tricks of the Trade, we're going to walk you through three very useful topics. First, Tiling in Fireworks using texture, then Aligning objects in Flash and finally, creating a custom error handler in PHP.

Eye-catching effects

The first subject, Tiling in Fireworks using texture, will be a huge time-saver for you. If you've ever had to create a repeating pattern for a graphic and had to do it by hand, you'll know exactly what we're talking about. We'll walk you through how to create the pattern and then use it on brush strokes or shapes.

As we discovered, not only does this method allow you to easily fill a space with a lovely repeating pattern, you can create some other funky effects that will catch the eye of your site visitors.

Snap to it

In our second subject, Aligning objects in Flash, you'll learn all about the three types of alignment you can use in Flash. These three types are Object Snapping, Pixel Snapping and Snap Alignment.

This is important if you want to create interfaces and graphics in Flash that are lined up exactly as you'd like. This snapping effect is tremendously helpful as it saves you having a lot of fiddly zooming in to align things by hand.

Elegant hacker prevention

Our third subject, Creating a custom error handler in PHP, will be a perfect solution for those developers out there, who want that extra bit of elegance and security in their website or web application. After you've read this walkthrough, you'll understand how to instruct PHP to use a custom error handler function instead of PHP's default

behaviour of spitting out errors to the screen. The reason why this is so important is because hackers purposefully try to cause errors on your website. The error messages are useful to the hackers as they are full of valuable information about the way your server or website is set up. By choosing to have these errors emailed to your system administrator (as we'll show you), you will be able to prevent hackers from gathering this sensitive information.

In addition to thwarting hackers plans, hiding these errors will also calm the nerves of your less techie visitors, errors are full of technical jargon that only scares visitors, instead of helping them.

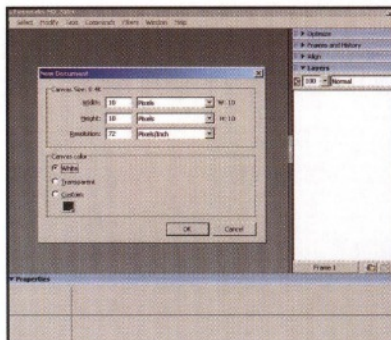
So, all in all, you'll find that after reading this instalment of Tricks of the Trade, you'll be armed to the teeth with valuable tips that will save you time, patience and make your website more robust and secure. ■

Part 1: Tiling in Fireworks using Texture

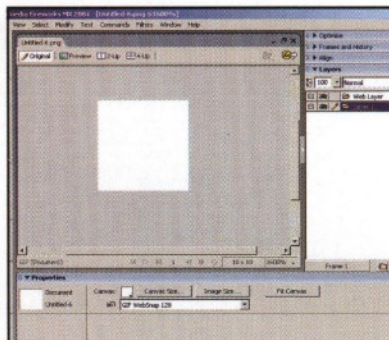
How to make a tiling pattern without having to hand-draw it...

Hiding CSS Styles

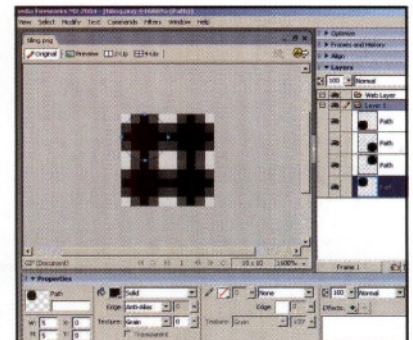
If you have CSS styles that are not compatible with Navigator 4.x, use the @import command instead of LINK.



1 If you'd like to make a tiling pattern without the painstaking process of hand drawing it, Fireworks has a sneaky way of doing it. Start by creating a new 10x10 pixel document with a white canvas.



2 Zoom into the canvas in order to draw your repeating pattern. Keep in mind that the canvas can be any size, as long as it is a square with sides of equal length.



3 Now go ahead and draw your pattern. We're going to use four circles of equal size. The colour of the shapes doesn't matter; when you use this pattern to draw, you'll choose the colour of the brush.



Indepth – Easy debugging with Java

When you're using any programming language for websites, such as PHP, ASP, ColdFusion etc, something will always go wrong (as many of you know). Little bugs pop up in the most annoying places and sometimes it seems impossible to find out where the problem is coming from.

Many web developers and designers resort to what we call debug hacking, which is a frantic methodology where you start printing out values of variables like a mad man, hoping to find the problem. We'd suggest going about it an entirely different way. The best way to deal with bugs in your web application or site is to actually plan on them appearing and build functionality into the code that allows for easy debugging which can be turned on or off. A great way to achieve this is by using JavaScript.

We'll go through some of the code here but the complete code can be found on the Cover Disc. Let's start by creating a new file called debugFuncs.php. This is where the debugging functions will reside. We'll need four functions: debugVar(), _debugOpenWindow(), _debugCapturePrintR() and _debugColorizeString(). The only function you'll actually be calling is debugVar, which in turn calls the other functions. Here is the code:

```
function debugVar ($sName,
```

```
$aData) {
    if (DEBUG) {
        _debugOpenWindow();
        $sCaptured = explode("\n", _debugCapturePrintR($aData));
        print "<script
        language= JavaScript >\n";
        print "debugWindow.
        document.writeln
        ( <b>$sName</b> );\n";
        print "debugWindow.
        document.writeln( <pre> )
        ;\n";
        foreach($sCaptured as
        $sLine) {
            print
            "debugWindow.document.
            writeln(
            "._debugColorizeString
            ($sLine)." );\n";
        }
        print "debugWindow.
        document.writeln( </
```

```
pre> );\n";
        print "self.focus();\n";
        print "</script>\n";
    }
}
```

Now, if you need to print out a variable, just call the function like this:

```
debugVar ( First Name ,
$sfirstName);
```

This will open up a JavaScript debug window and print out the variable's value for you. Lastly, we need to ensure that debugging can be easily turned on or off. Create a new PHP file called config.php that is included at the top of every page. In this file, we're going to use a constant that turns debugging on or off. Add the line:

```
define ( DEBUG , 0 );
```

Now just change this value to 1 to turn on debugging or 0 to turn it off.

Frame rates in Flash

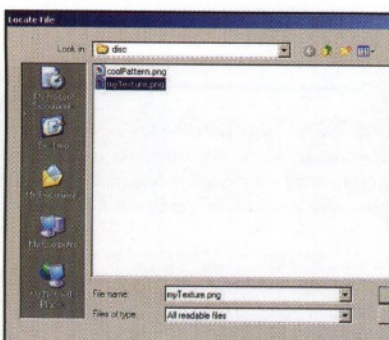
If you need your Flash movie to be smoother, increase the frame rate. Keep in mind, though, that this will make it sluggish on very old computers.

Avoiding CSS shrinking text

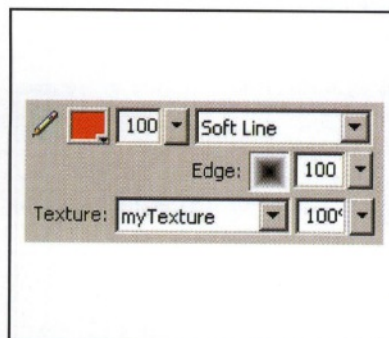
Be careful when combining font-size with a percentage value (i.e. 80%) and nested tables, as the text could become unreadable.

Fireworks and Flash

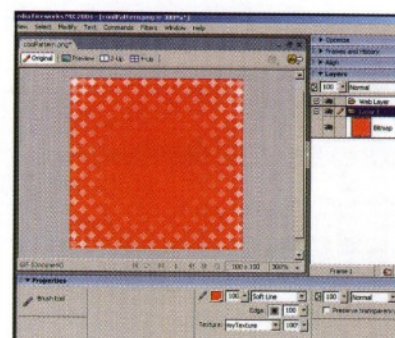
Copying and pasting between Fireworks and Flash has been improved with MX 2004. Just select the item in Fireworks, copy it, then open Flash and paste it.



4 Save this document as myTexture.png. Open up a new document. Choose the Brush tool (B) and open up the Properties Panel (Window Properties). On the Properties Panel, click the dropdown for Texture, scroll down to Other and browse for myTexture.png.



5 On the Properties Panel, make sure 100% is chosen next to the Textures dropdown menu. Have a play with the Edge value, stroke type and colour. You can achieve some really funky results.



6 You can also use textures on Fills if you'd like. Simply draw a shape and choose your texture on the Properties Panel. However, in our opinion, the Brush tool is much more effective.



Part 2: Aligning objects in Flash

Snapping to a grid...

Flash timeline folders

In order to organise your layers in Flash and to make them manageable, create folders. Choose Insert Folder Layer and drag your layers onto the new folder.

Trouble with CSS being displayed?

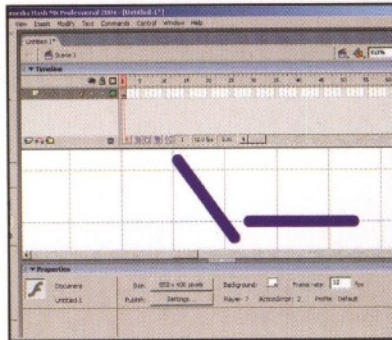
If you're sure your CSS is correct and it's still not working, make sure your ISP has the .css file extension mapped to MIME type text/css.

Flash layer properties

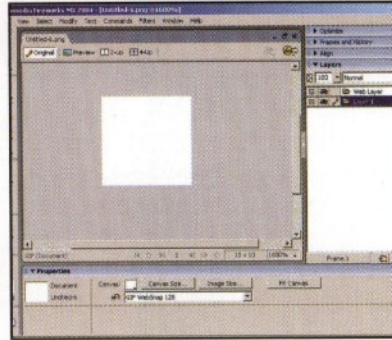
If you xFlash gives you extraordinary control of your layers. Right-click (Ctrl-click on Macs) on the layer and have a play with the properties.

Fonts in Flash

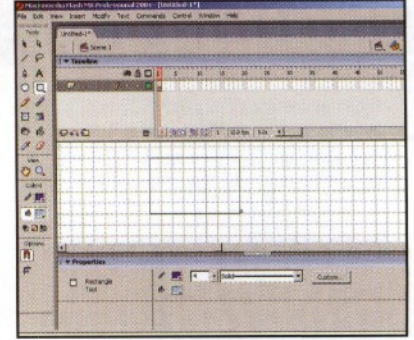
To make sure your fonts display properly in Flash, choose Character... on the Properties Panel and choose the set you'd like. This bumps up file size, so be careful.



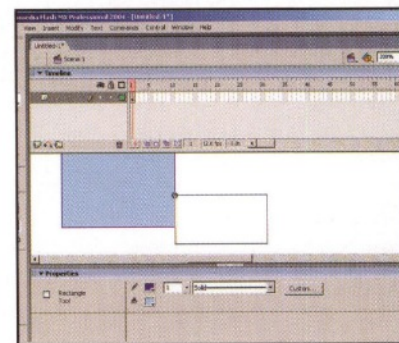
1 When using Flash it is often necessary to make sure things are lining up correctly. Most programs call this snapping or snapping to a grid. Thankfully, Flash also includes this capability.



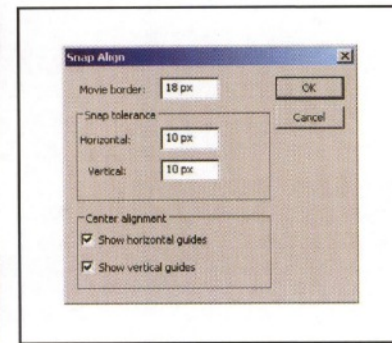
2 Flash provides three ways for you to align objects on the Stage. The first is Object Snapping which lets you snap objects directly to other objects along their edges. Select View > Snapping > Snap to Objects.



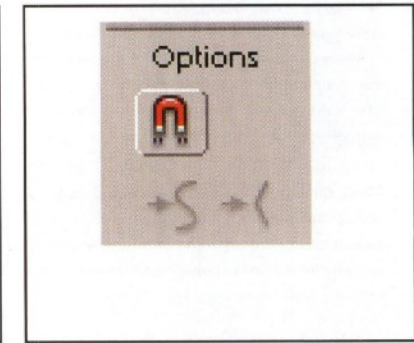
3 Secondly, Pixel snapping. This lets you snap objects directly to individual pixels or lines of pixels on the Stage. Select View > Snapping > Snap to Pixels.



4 Thirdly, is Snap Alignment. This lets you snap objects to a specified snap tolerance, a preset boundary between objects and other objects or between objects and the edge of the Stage. Select View > Snapping > Edit Snap Align.



5 While editing the Snap Alignment settings, you can set the snap tolerance between horizontal or vertical edges of objects, and between objects' edges and the Stage border. You can also turn on snap alignment between the horizontal and the vertical centres of objects.



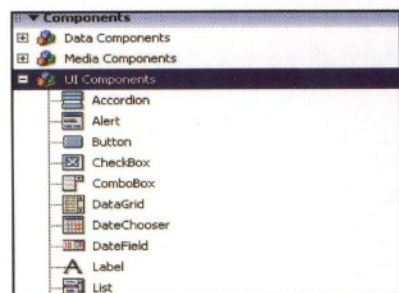
6 Remember that all Snap Alignment settings are measured in pixels. The Flash Tools Menu has a shortcut button to turn on Object Snapping. Just click the magnet icon on the Tools menu to toggle Object Snapping.

Part 3: Pre-built Flash components

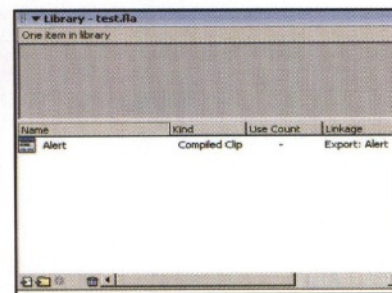
Using the Alert component...

Plesk

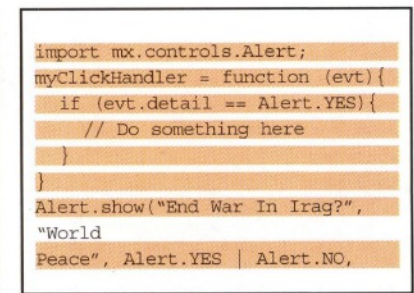
If you're splashing out on a Linux dedicated server, make sure you get Plesk (www.sw-soft.com/en/products/plesk) installed. It's a brilliant piece of software that will enable you to run your server without being a pro at Linux.



1 Flash MX 2004 Pro has a huge amount of useful pre-built components for you to use. One of those is the Alert Box. Start off by opening the Component Panel (Window > Development Panels > Components). Expand the UI Components by double-clicking it.



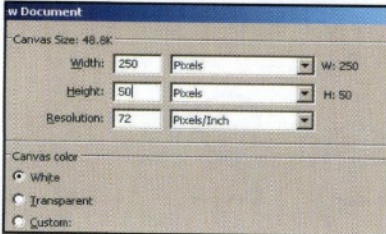
2 Double-click the Alert component. This adds it to the Stage and the Library. We don't want it on the Stage right now, so select it and hit Delete. Now we need to add the ActionScript to make the Alert window work. Select the first frame on the time line and open the Actions Panel (F9).



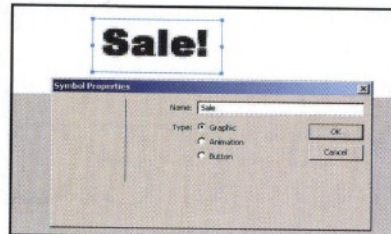
3 Flash needs a handler in order to react to user input and respond in the correct way. By using some fairly simple code, we can tell Flash exactly what to do when the user chooses an option from our alert box. After we've written the handler for Flash, the code that controls the Alert box is fairly simple.

Part 4: Fireworks again

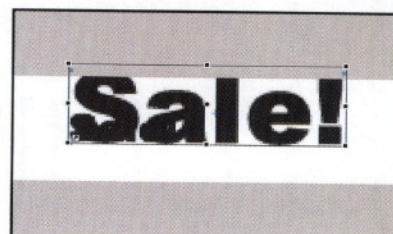
Creating a simple animated GIF...



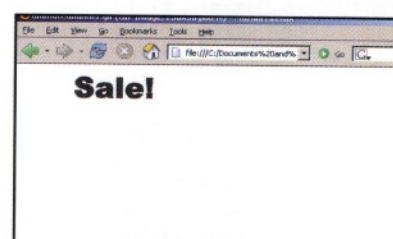
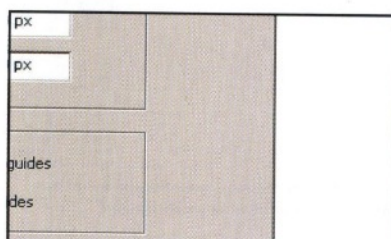
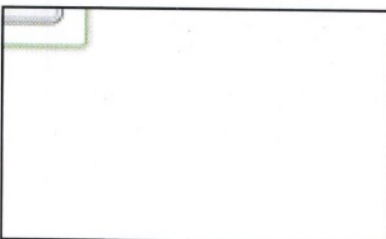
1 You will often need to create animated GIFs for banners on web sites. It's really quite simple. Start by opening Fireworks MX and create a new document (File > New), with height of 50 and width of 250.



2 As an example, we're going to make the word Sale! grow larger and larger in the image. Choose the Text tool (T) and type Sale!, using Arial Black, size 18. Select the Pointer Tool (V), then select Modify > Symbol > Convert to Symbol. Type Sale into the Name field, select Graphic and click OK.



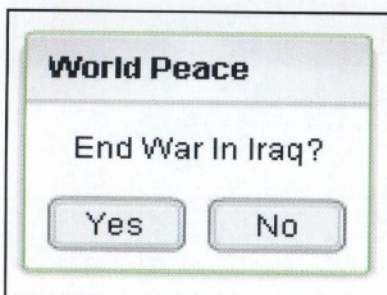
3 We want Sale! to grow larger. The technical word for this is tweening. Select the Sale! Symbol and then copy and paste it. Select the copied Sale! symbol, select the Scale Tool (Q) and drag one of the corners of the symbol.



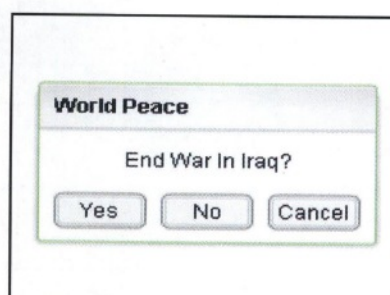
4 Move the enlarged symbol back to the centre of the canvas. Now select both symbols and choose Modify > Symbol > Tween Instances. Leave the steps at ten and click OK. Open the Frames Panel (Window > Frames) and you'll see how Fireworks will spread the animation out over multiple frames.

5 Choose File > Export Preview to save the animated GIF. Choose Animated GIF from the Format dropdown. You can click the Play button to get a preview. If you're happy with it, click Export.

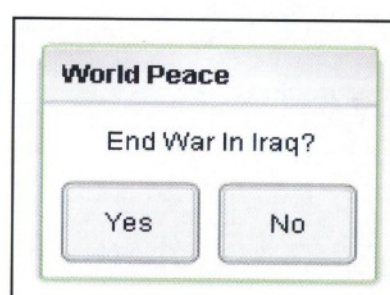
6 The final result is a simple animated GIF that can be used as a banner on your site. If you want it to be slightly smoother, increase the number of frames. If the size of the GIF is too large, reduce the frames.



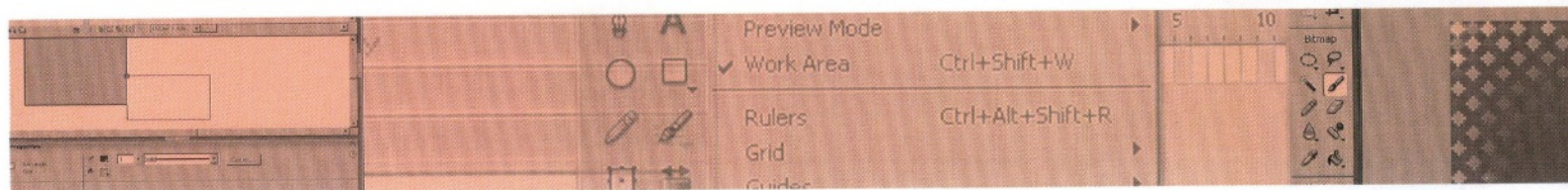
4 Here's how Alert.show works: the first argument is the alert message, the second is the title of the alert box, the third represents buttons we want (separated by a | character), the fourth is the name of the parent and the fifth is what the function should be called once a button is clicked.



5 For Yes, No and Cancel buttons, the call to Alert.show is like this:
`Alert.show("End War In Iraq?", "World Peace", Alert.YES | Alert.NO | Alert.CANCEL, this, myClickHandler);`



6 You can customise the way the Alert box looks by changing the properties. To change the width and height of the buttons, simply add these two lines above the call to Alert.show:
`Alert.buttonHeight = 40;
Alert.buttonWidth = 60;`



Tricks of the trade 2

More handy web tricks and design tips from **Ryan Carson** to help make your web site even more superb...

Browser compatible

Make sure to always test your website on multiple browsers like Safari, Firefox, Netscape and IE.

Notepad

On a PC, Notepad is a useful ally. We often use it as a cut & paste board where you can store little bits of HTML or code for future use.

This time in Tricks of the Trade we're looking at some of the cool things Fireworks can do. For creating or editing graphics, Fireworks is one of the best tools available. It's not cheap at £250, but it's an investment that will reward you.

In the past, automatic code generating software always produced horribly messy and clunky code. Recently, however, Macromedia has made a real jump in creating software that does a decent job in auto-writing code for you. Of course, we'd always recommend you check the code and make sure you understand it but, on the whole, it's a powerful tool that you should think about using. Fireworks calls this automatic code generation Behaviors.

A great example of this is using Fireworks to write simple JavaScript that will swap images on mouse rollover. Create two slices of your image. Right-click one of the slices



Using Fireworks to swap images is just one of its useful solutions for editing web graphics

and choose Add Swap Image Behavior from the pop-up menu. Follow the on-screen instructions, choose the slice that should be used for swapping, then hit OK to finish. You'll notice that there is now a blue line, when one of the slices is selected, that connects one slice to the other. This is the way that Fireworks tells you that there is a Fireworks Behavior attached to the slice. Now choose File > Export and choose an



It's a simple matter to take different slices from one image and use them to create an effective rollover

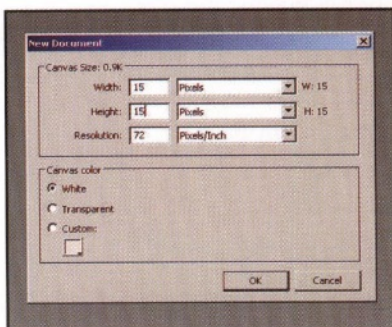
HTML filename and Fireworks will take care of writing the JavaScript for you. This is just one of many examples that shows how software has really come a long way in helping you create powerful solutions in a very short amount of time. ■

Part 1: Creating status bar messages in Fireworks

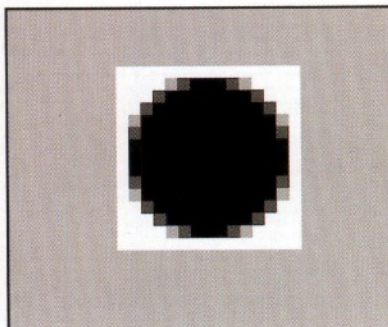
Fireworks is great for creating web content...

Quick images

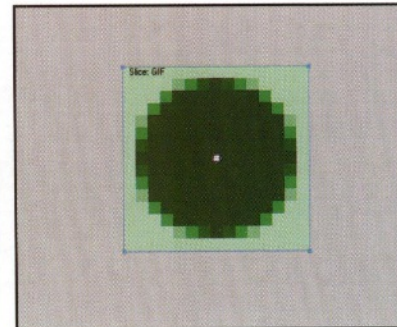
In Fireworks, after slicing up an image, you can export with one click. Simply right-click on the slice and choose Export selected slice.



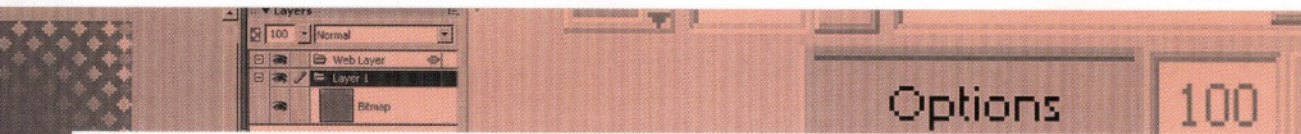
1 We're going to show you how to create a message in the browser status bar by mousing over a graphic. Start by creating a new canvas, 15x15 pixels in dimension.



2 Zoom into the canvas and create a shape. You can use any shape, colour or size you want – we've created a 13x13 pixel black circle.



3 Next, use the Slice Tool (K) and draw a slice that covers the whole graphic. You'll notice that your slice has appeared in the Web Layer folder in the Layers Panel.



Indepth – Object Oriented Programming

Object Oriented Programming (OOP) is the practice of breaking up code into logical parts that each serve a separate purpose. This practice usually leads to code that is more reusable and much easier to maintain. Most classes will relate to objects stored in a database. With PHP, you'll usually be using a MySQL database. If you want a good book to help you get started on the basics, you could have a look at Sam's Teach Yourself PHP in 24 Hours, Third Edition. The first step to delving into OOP is a clear understanding of what a class is and what an object is. A class is a set of data that has something in common. A good example would be an article on a news web site. An object is an instance of that class. For example, an object of the news class would be a news story on the opening ceremony at the Olympic games.

Each class has variables and functions that relate to it. For instance, a `news_article` class could have the following variables: `id`, `headline` and `story`. It might have the following functions: `create_article`, `delete_article`, `print_article`, etc. Here's the basic structure of a PHP class:

```
class news_article {
    var $id;
    var $headline;
```

```
    var $story;

    function news_article($id) {
        // get article from
        database
    }

    function print_article() {
        // outputs article to
        screen
    }
}
```

As you can see, there is a function inside the class called `news_article`. This is called the constructor. Every class has one and it is responsible for any initial setup or data retrieval. Now that you are familiar with the code for this particular class (usually stored in a file called `news_article.class.php`), we are going to show you how to use this it:

```
// Print article ID 1
$my_article = new news_
article(1);
$my_article->print_article();
// Print article ID 2
```

```
// Print article ID 1
$my_article = new news_
article(1);
$my_article->print_article();
// Print article ID 2
```

Learning the differences between the class and the object in OOP might take a while but it's worth the work

```
$my_article = new news_
article(2);
$my_article->print_article();
```

That's pretty much all there is to it. We know it takes a bit to get your head around OOP initially, but once you do, you'll be glad you did.

Sam's Teach Yourself PHP in 24 hours is an easy access point for keen learners



PDF web pages

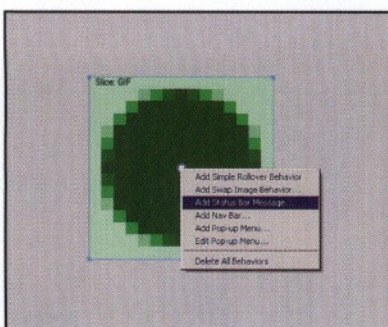
Adobe Acrobat has the ability to download entire web sites and convert them to PDF. Just choose File > Open Web Page

Fill Transform Tool

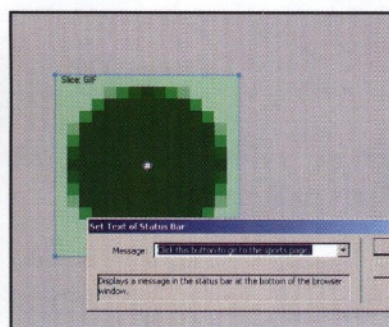
In Flash, when you use a gradient or bitmap to fill a shape, you can use the Fill Transform Tool to adjust the fill to your exact specifications.

Email links

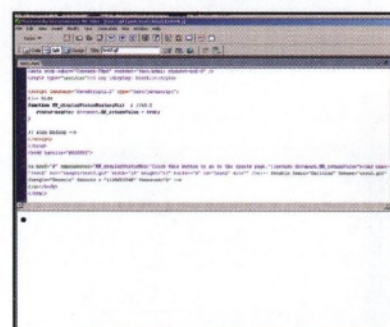
A handy tool for generating complex mailto HTML email links can be found at developer.netscape.com/viewsource/husted_mailto/mailtoApp.html



4 Choose the Pointer Tool (V) and select your new slice. You'll see a crosshair button appear in the middle. Click that once and choose Add Status Bar Message.



5 Type the message that you would like to appear when users roll over this graphic and hit OK. This tells Fireworks to create JavaScript that will trigger this behavior.



6 Choose File > Export from the menu and hit Save. Now open up the HTML page Fireworks created and you'll see it's placed the graphic and written the JavaScript for you.



Part 2: Creating an HTML pop-up menu with Fireworks

Fireworks can write some powerful JavaScript automatically for you...

Colour coordination

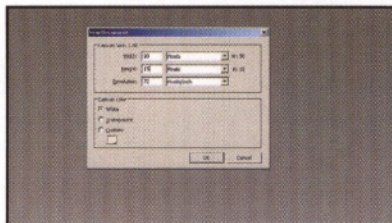
Behr has created an ultra useful colour tool that helps you find coordinating colours. [w] www.behr.com/behrr/workbook

Command line downloads

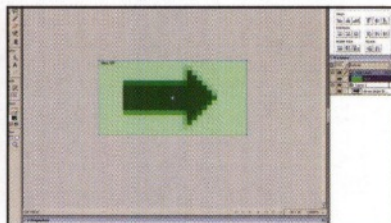
Wget is a command line program that's useful for downloading content from HTTP and FTP servers. Grab it from [w] www.qtm.net/~twegscheid/wget.html

Automated tasks

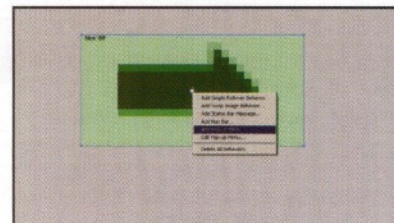
Would you like a certain application to be run every hour, minute, day, etc? Cron [w] www.kalab.com/freeware/cron/cron.htm is a very handy tool for Windows users.



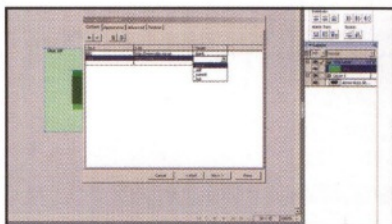
1 We're going to show you how to use Fireworks and JavaScript to create a handy pop-up menu. Open Fireworks and create a new canvas 30x15 pixels.



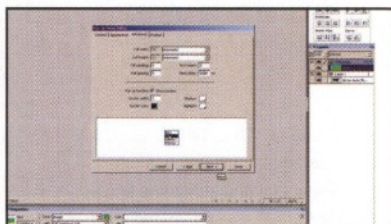
2 Add an arrow shape (hold down the Rectangle Tool button and choose the Arrow shape) to the canvas. The colour doesn't matter. Now choose the Slice Tool (K) and add a slice.



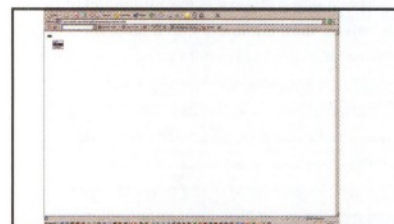
3 Select the slice with the Pointer Tool (V) and a crosshair button will appear in the centre. Click that button and choose Add Pop-up Menu, which should bring up the Pop-up Menu Editor.



4 On the Content tab, click the space under Text and enter BBC then hit the Tab key, enter [w] www.bbc.co.uk, hit Tab again and choose _blank from the dropdown. Repeat to add more links.



5 For the Appearance, Advanced and Position tabs, enter values to suit the visual appearance of your menu. It gives you a preview of the values you've chosen. When you've finished, click Done.



6 Now choose File > Export and type a name for your HTML page. Hit Save and Fireworks writes all the JavaScript for you.

Tutorial 3: Creating true black and white images

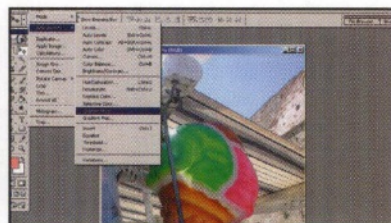
You can cheat, or you can create black and white images the right way...

Instant Answers?

GuruNet apparently gives you answers, instead of links, from your search results. Give it a try at [w] www.gurunet.com



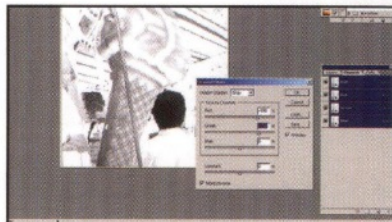
1 When creating black and white images, people often cheat by choosing Image > Mode > Grayscale in Photoshop but this means you have little control over the final look of the image.



2 Instead, we're going to show you a handy method that gives you maximum control. Choose Image > Adjustments > Channel Mixer.



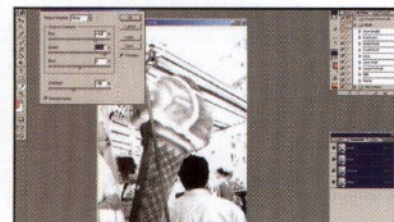
3 Tick Monochrome on the bottom of the panel and the image will become black and white. You'll notice that the only Output Channel available in the dropdown menu is Gray.



4 Have a play with the sliders. You'll see how changing Red, Green and Blue will give you interesting results.



5 If your computer is running slowly, you can uncheck Preview but if your computer can handle it, it's a great way to see exactly how your changes affect the image.



6 Finally, you can have a play with the Constant value. This option adds a black or white channel of varying opacity – negative values act as a black channel, positive values act as white.



● Subscribe now and get a coverline-free cover, delivered directly to your door

● To subscribe call 0870 8374722

COMPUTER arts

THE WORLD'S BEST-SELLING CREATIVE MAGAZINE



NOW AVAILABLE IN DIGITAL FORMAT

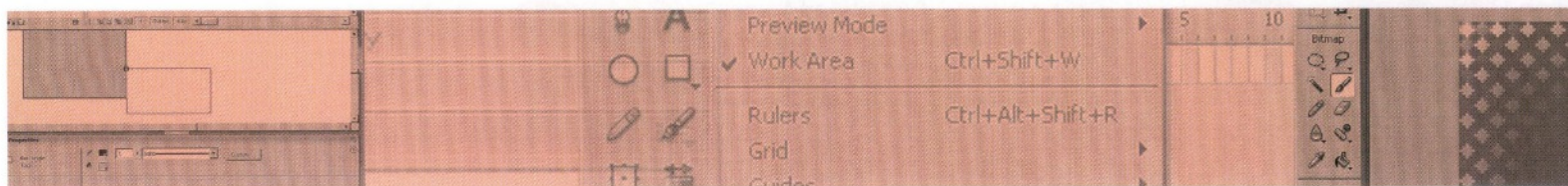


● Subscribe to the world's best-selling creative magazine

● Receive 13 issues for just £50 (by direct debit)

● Call 0870 8374722

TO SUBSCRIBE TO COMPUTER ARTS, VISIT **WWW.COMPUTERARTS.CO.UK** TODAY



Tricks of the trade 3

Improve your designs with **Ryan Carson's** expert web-building tricks

Flash — Adjusting curves

By using the Subselection Tool (A), you can readjust the points on a vector line after you've drawn them.

Websites — Word of mouth

The best way to get people to go to your site is by word of mouth. Get onto forums and start asking people what they think of it.

With the advent of Flash MX 2004 Professional, users are equipped with a powerful array of tools that come pre-packaged with Flash. A great example of this is the Components that come with Flash. Many of these have been designed to reduce the time it takes to get your Flash file built and live on the web. In the past, Flash designers/developers would spend ages doing the same task over and over again. Now all you have to do is drag the appropriate Component from the Component Panel onto your Flash Canvas, and away you go!

Probably the only downside to using pre-built Flash Components is that it is often difficult and complex to modify the way they look. If you're determined, you can drill down into the movie clips and modify them as necessary. However, you could inadvertently break the Component, rendering it useless. So if you're brave, and have a fair bit of

ActionScript knowledge, have a bash. If not, it's best to use the Components in their default state.

Let's go over a couple of the basics now. Start by opening up Flash MX 2004 Professional. If you don't have a copy, you can get a 30-day free trial from the Macromedia site at www.macromedia.com/flash. Open up the Components Library (Window > Development Panels > Components) and have a browse down the list. You've got an amazing list that includes things like: Alert, which is similar to a JavaScript alert box; ComboBox, which is like an HTML dropdown combi box; RadioButton, which, as you'd expect, is like the HTML radio button in a form; TextArea, which is a scrollable text area (we cover this in one of our tutorials below) and many many more!

Let's try one out. Drag the Button Component from the Component Panel onto the Canvas. Once it is on the Canvas, select

it. Then open the Properties Panel (Window > Properties). You'll see two tabs, Properties and Parameters on the Properties Panel. Select the Parameters tab, enter the button text into the label area. Now hit Q on your keyboard and use the resize handles to make the button fit your text label.

Now to make the button functional, we need to add a bit of ActionScript. Select the button and give it the instance name (on the Properties Panel) btn_Buy. Then select the first frame of the timeline and add the following code:

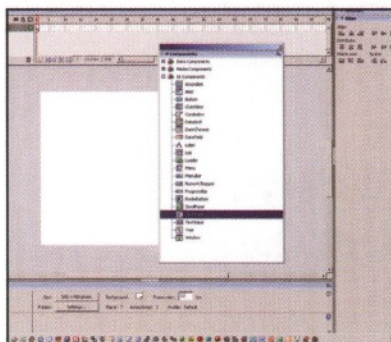
```
btnListener = new Object();
btnListener.click = function
(evt){
    // Do action here
}
btn_buy.addEventListener("click"
, btnListener);
```

Part 1: Scrolling text

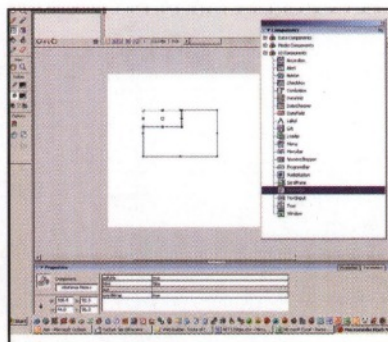
How to make scrolling text in Flash...

Flash — Making lines smooth

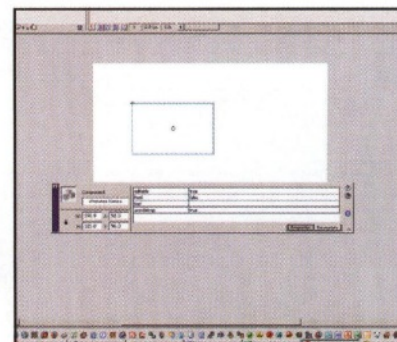
Draw a line with the Pencil Tool and choose Modify > Shape > Smooth (or Straighten). This is a quick and easy way to clean up messy lines.



1 Open Flash and create a new document. Select Layer 1 and open the Components Panel (Window > Development Panels > Components). Components in Flash MX 2004 are very powerful.



2 Drag the TextArea component onto the canvas from the Components Panel. Resize it to your preferred size by hitting Q and then drag the handles to whatever height and width you wish.



3 With the TextArea component selected, open the Properties Panel (Window > Properties). Choose the Parameters tab on the lower right side of the Properties Panel.



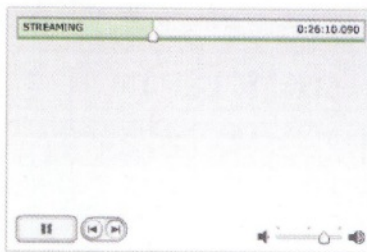
In depth – Adding an MP3 player

We're focusing on Flash here, and Flash MX 2004 Pro has some crazy powerful features. The Media Components are amazingly potent and useful. You can literally drag-and-drop the components in your Flash file and they perform amazingly complex functionality. If you haven't got a copy of Flash MX 2004 Pro, just head to www.macromedia.com/flash. Although it will only be actually worth buying if you've got paying clients who are asking for advanced Flash programming, it's still a lot of fun to play with. Especially if you've got your hands dirty in Flash before. The Component we're going to be focusing on is the MediaPlayer Component. This is a powerful piece of code that the Macromedia team has coded and included in with Flash MX 2004 Pro. It allows you to include a MP3 or video in your Flash movie and

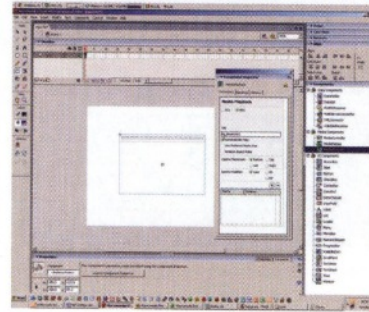


Media components – the hard work's done for you!

have a controller that can be used to start, stop, play, pause, etc. Start by opening Flash and creating a new document. Open up the Components Panel (Window > Development Panels > Components). You'll see three groups: Data Components, Media Components and UI Components. Drag the MediaPlayer Component onto the Flash Canvas. On the Properties Panel (Window > Properties), click Launch Component Inspector on the Parameters tab. Place your MP3 file in the same directory as this Flash file. In the URL box, enter the name of the MP3, for instance `big_pimpin.mp3`. Leave the rest of the options unchanged. Make sure the MP3 radio box is chosen at the top of the inspector panel. If you feel adventurous, you can try different



The MediaPlayer Component gives complete control



The Component Inspector – helpful for customising components. Here we are creating an MP3 player.

options for the Control Placement and Control Visibility. Now give it a test by choosing Control > Test Movie. You'll see that the controller has a really nice, minimalist appearance. Now mouse over it. The controller expands into a proper play head! Now just upload both the SWF and MP3 file to your website (in the same directory) and you're set to go.



A quality control that's sleek, sexy and easy to use

Flash – Keeping music in sync

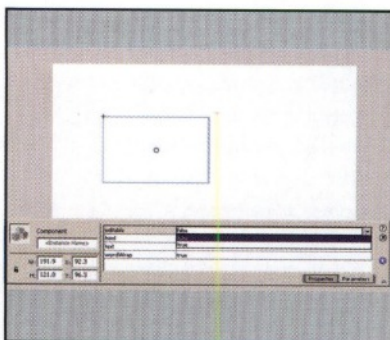
When using music in your Flash movies, make sure to set the audio options to Stream and Flash will drop frames as necessary to keep the animation and music synchronised.

HTML – Don't use Word

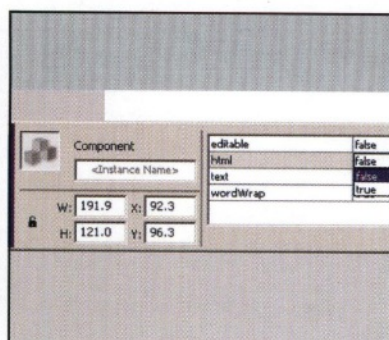
Microsoft Word can be used to create HTML pages by selecting File > Save As Web Page. This creates messy, horrible HTML. It is best to avoid this feature.

Flash – Making star shapes

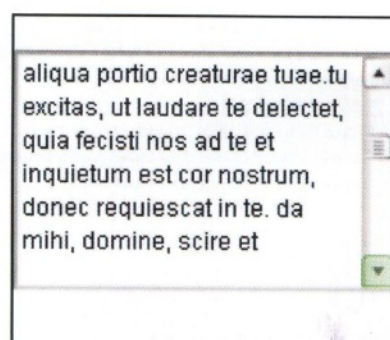
Click and hold the Rectangle Tool on the Tools menu and choose PolyStar Tool. You can adjust the settings in the Properties Panel.



4 There are four fields: Editable, HTML, Text and Wordwrap. Generally choose False for the editable dropdown option. This will keep your visitors from changing the text.



5 Enter False for the HTML dropdown option. If you choose to have HTML tags in your text, then change this to True. Your HTML tags will then be used by the Flash text area.



6 Lastly, copy your text into the text field. This can be as long as you like and Flash will add a scrollbar if it is necessary. It's like magic!



Part 2: Non-aliased text

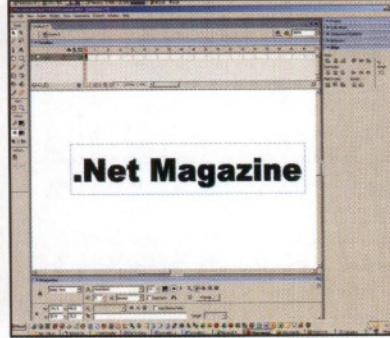
How to create non-aliased text in Flash

Flash — Snapping objects

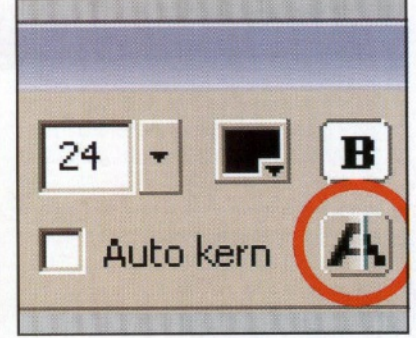
A great way to keep things lined up in Flash is to use Snapping. Try it out by selecting one of the options under View > Snapping.



1 In the old days it was very difficult to keep Flash from anti-aliasing your text – making the edges smooth. Getting that pixel effect on your text was a real pain in the backside!



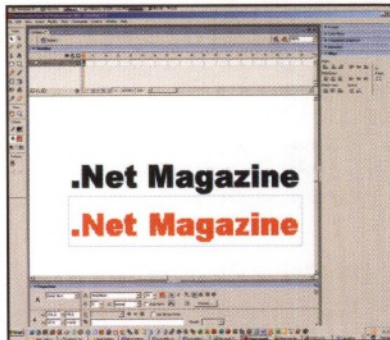
2 Start off by opening Flash and creating a new document. Choose the Text Tool (T) and type something onto the canvas. Anything will do for now. Make sure Static Text is chosen on the Properties Panel.



3 You'll notice a new button on the Properties Panel that wasn't there for text in Flash MX. It's called the Alias Button.

Websites — A second opinion

Whenever you finish a website, always ask your friends and workmates to tell you what they think. This can be a great way of improving the site before launching it live.



4 After typing your text, click the Alias Button and watch what happens. Your text suddenly loses its anti-aliased edges!



5 The best way to use this feature is when you're attempting to make text in Flash look like HTML text on a web page. Try choosing Arial or Verdana size 12.



6 Give the movie a test (Control > Test Movie) and you'll see the amazing result. Before this feature, Flash designers had to use specially made Pixel Fonts to achieve this effect.

Flash — Keep it organised!

Flash Libraries can get messy in no time at all. To avoid this, create Folders in your Library and store symbols inside them.

Websites — Measure your success

Make sure to measure your hits on your website, otherwise you won't know if it's successful. Two great affordable products are FastCounter Pro (tinyurl.com/5za3s) and Hitbox Pro ([w]www.hitboxprofessional.com).

Using JavaScript to check a form

Often, it is tremendously useful to use JavaScript to check if an HTML form has been submitted correctly. In this example, we show you how to check if the user has entered text into a Name text field. The idea here is that the form is only submitted if the JavaScript function returns true. If not, it creates an Alert box.

```
<html>
<head>
<script language="javascript"
type="text/javascript">
<!--
function
validRequired(formField) {
```

```
var result = true;
if (formField.value == "")
{
alert( Please enter a
value for the Name
field. );
formField.focus();
result = false;
}
return result;
}
// -->
</script>
</head>
```

```
<body>
<form action="text.html"
name="form" method="post"
onSubmit="return
validRequired(document.form.
name)" enctype="text/plain">
>
</form>
</body>
</html>
```


COMPUTER arts projects

EXCLUSIVE COMPETITION: WIN A TOP-OF-THE-RANGE HP A1 DESIGNJET PRINTER WORTH £1292!

COMPUTER arts projects
THE IN-DEPTH GUIDE FOR DIGITAL CREATIVES

**EVERYTHING YOU NEED TO BECOME A...
FREELANCE DESIGNER!**

- Win big-name clients
- Build a top online portfolio
- Nail your first showreel
- Design business stationery
- Unravel tax and copyright

PLUS
TOP FREELANCER SECRETS
Miles Donovan, Liselotte Watkins and illdesigners reveal all

40 SELF-PROMOTION TIPS*
Nick and Adam Hayes of Identikal on how to conquer the design industry

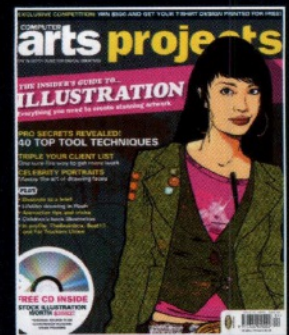
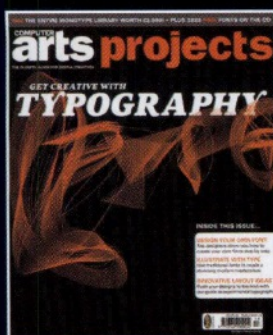
FREE CD INSIDE
PROTECT YOUR WORK!
WITH (WATERMARK) FULL SOFTWARE

BUSINESS MANAGEMENT TOOLS
20 FREE STOCK IMAGES
TUTORIAL SOURCE FILES

ISSUE 71 MAY 2006 £6.95
9 771740 500004
UK edition - Printed in the UK



Computer Arts Projects is the only in-depth guide for digital creatives to cover a different subject every month...



**SUBSCRIBE TO COMPUTER ARTS PROJECTS TODAY AT
WWW.MYFAVOURITEMAGAZINES.CO.UK OR CALL 0870 837 4722**



COMPUTER INC.
© 2000

60 ACTIONSCRIPT TIPS

Whether you're a novice or an expert, you're sure to find something in the next few pages to use in your next ActionScript project...



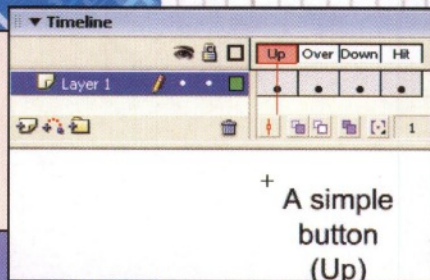
You're about to find out things you can do in Flash that you'd never have thought possible, whatever your skill level. These tips are split into three categories: Beginner, Intermediate and Advanced. The Beginner section is for those just starting out with ActionScript. You'll find tips on using buttons, making objects move, using functions and arrays and even a tip on how to tell if your movie is running on a PC or a Mac. If you've been using ActionScript

for a little while and know the basics, the Intermediate tips will be more useful for you. They'll enable you to create a text field on the fly, open a file requester dialog and set a timeout event. Finally, the Advanced tips are for those who know their ActionScript inside out, but may not have become familiar with custom classes, the LocalConnection object and XML news feeds. It's well worth reviewing all the tips, because even the advanced advice is fairly easy to add to your own code.

60 ACTIONSCRIPT TIPS

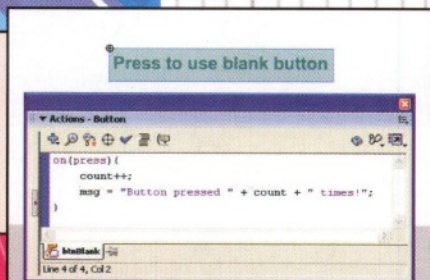
TIP 1

You can create a button using a MovieClip comprising four frames.



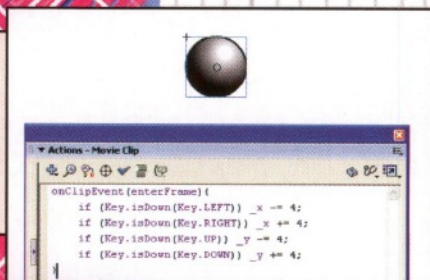
TIP 2

A blank button is the simplest type to create, using just one frame.



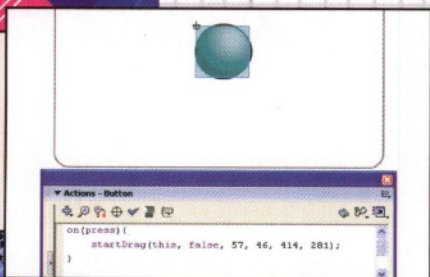
TIP 5

Add this simple code to move objects with the arrow keys.



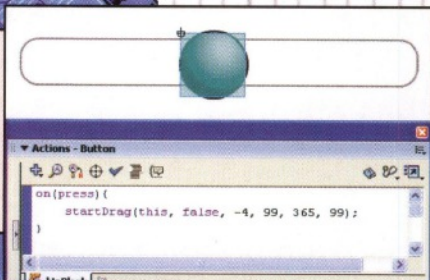
TIP 7

Limit the drag region of an object by limiting values to the call to startDrag.



TIP 8

You can create a simple scroll bar by limiting the y values.



1. Creating a button

Buttons are just a special kind of four-frame MovieClip. The four frames are the up, over and down images and the hit area, in that order. You can add a script to these frames by clicking the button and adding events for press, release, rollover and rollout.

2. Using a blank button

The simplest kind of button you can create contains only the fourth frame: the hit area. A blank button is displayed in turquoise at design time, but is blank at runtime. You can simply place them over the top of other objects to create a clickable button.

3. A multistate button

Sometimes you'll need a button that can be set to multiple states. To do this, create a button, select it, then press F8, and create a MovieClip symbol from the button. Now when the button is clicked, you'll be taken to a new frame inside the clip.

4. Jumping with a button

Often a button will be used to navigate a movie. Simply add the following code:

```
on (press) {
    gotoAndStop ("MyMarker");
}
```

This causes the playback of the timeline where the button is placed to jump to the frame labelled MyMarker.

5. Moving with the arrow keys

To move an object on screen it should be a MovieClip. You can add code to the MovieClip by adding an onClipEvent handler. In this example, we add an onClipEvent (enterFrame) handler. We check the keyboard using the Key object. If the arrow keys are pressed, we alter the position of the MovieClip by adjusting the _x and _y values.

6. Moving by dragging

Create a MovieClip with a blank button inside. Add the following code to the button and you'll have a draggable clip:

```
on (press) {
    startDrag(this, false);
}
```

```
on (release) {
    stopDrag();
}
```

7. Limiting the drag region

Often you'll want to limit the region in which the dragged object can move. This is done by adding four additional parameters to the call to startDrag: left, top, right and bottom values. To find these values, use the following code to display the current position of your clip:

```
trace(_x + ", " + _y);
```

Write down the values at the top-left and bottom-right corners and use these in the call to startDrag.

8. Simple scroll bar

You can use the draggable clip to create a simple scroll bar. Just limit the y values to the same value for top and bottom and you have a horizontal scroll bar. You can use the _x value to give a value for the scroll bar using the following code, where left and right are the values used in startDrag, and limit is the maximum value:

```
value = ((_x - left) / (right - left)) * limit;
```

9. Link the scroll bar to a text box

To display the value, create a text box. Use the Properties window to set it to Dynamic and enter a variable name, in this case 'scroll.value'. 'scroll' is the instance name for the scroll bar. The value 'value' is set when the button is released.

10. Moving in a circle

To move in a circle, use the Math object methods sin and cos. If the radius of the circle is x, then the position of the object at angle theta is:

```
_x = x * Math.cos(theta);
```

```
_y = x * Math.sin(theta);
```

To centre this in the Movie, add half the width to the _x value and half the height to the _y value.

11. Creating a function

Functions are very useful for keeping a block of code in one place. Use the keyword function followed by your choice of name and then brackets. The code for the function is contained between curly braces.

```
function myfunction() {
    trace("Called myfunction");
}
```

You call the function simply by using the name of the function and brackets.

```
myfunction();
```


BEGINNER

You can pass parameters to a function using this code:

```
myfunction("Hello world");

function myfunction(str){
    trace(str);
}
```

12. Returning a value from a function

To return a value from a function, use the return keyword. In this example, the output from trace will be 16.

```
trace (sum(11, 5));
```

```
function sum(a, b){
    return (a + b);
}
```

13. Creating an Array

Arrays are useful for storing your information. You can create an Array using the new keyword:

```
var emptyArray = new Array()
var populatedArray = new
Array(1, 2, 3, 4, "a", "b",
false);
```

14. Accessing Array elements

```
var myArray = new
Array("Macromedia", "Flash",
"MX", "2004");
```

To access a member of myArray use myArray[i], where 'i' can be any value from 0 to one less than the number of elements in the Array.

myArray[0] is "Macromedia"
myArray[1] is "Flash" etc.
The number of elements in an Array is myArray.length.

15. Deleting elements

To remove an element from an Array at index i, use the following code, where deleteCount is the number of elements to remove:

```
myArray.splice(i, deleteCount);
```

You can use the same method to add elements by setting the value of deleteCount to 0 and then passing further parameters. In this instance, the element "Flash" will be added at index 1.

```
myArray.splice(1, 0, "Flash");
```

16. Multidimensional Arrays

```
var subArray1 = new
Array("Macromedia", "Flash",
"MX", "2004");
```

```
var subArray2 = new
Array("Macromedia", "Director",
"MX", "2004");
var myArray = new Array("Hello",
subArray1, subArray2);
```

Using the above myArray[0][1] is 'Flash': 0 to choose the first element of myArray and then 1 to choose the second element of the embedded Array.

17. Joining an Array

You can join an Array together and return it as a String, a line of text. Each element is separated from the previous one with the parameter passed to the method join.

```
var myArray = new
Array("Macromedia", "Flash",
"MX", "2004");
trace(myArray.join(" "));
```

In this instance, 'Macromedia Flash MX 2004' is sent to the Output Window.

18. Creating an Array from a String

```
str = "Macromedia Flash MX
2004";
myArray = str.split(" ");
```

Using the split method of a String object, you can create an Array. You pass the symbol used to break the String into separate elements - in this case, a space character. The result of this is an Array of four elements: "Macromedia", "Flash", "MX", "2004".

19. Variable scope

Flash can contain MovieClips embedded into MovieClips. The root level timeline is itself a MovieClip. If you have a variable at the root called x then this is different from a variable called x inside a MovieClip called mc.

To access variables in the MovieClip mc from the root use: mc.x

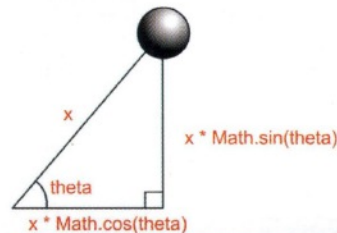
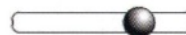
To access variables in the root from mc use: _root.x or this._parent.x

20. PC or Mac

To find out if the Flash movie is running on a PC or a Mac use:

```
os = $version;
if (os.substr(0, 1)=="W"){
    os = "Windows";
}else{
    os = "MAC";
}
```

73



Original array has 4

members

0: Macromedia

1: Flash

2: MX

3: 2004

Deleted element 1.

Array now has 3 members

Original array

0: Hello

1: Embedded array

0: Macromedia

1: Flash

2: MX

3: 2004

2: Embedded array

```
function dumpArray(n, str, tabs){
    var tabStr = "";
    for (var i=0; i<tabs; i++) tabStr += "  ";
    if (n.length>0){
        if (str=="") trace(tabStr + str);
        for (var i=0; i<n.length; i++){
            if (typeof(n[i])=="object"){
                trace(tabStr + i + ": Embedded array");
                dumpArray(n[i], "", tabs + 1);
            }
            else{
                trace(tabStr + i + ": " + n[i]);
            }
        }
    }
}
```

TIP 09

You can link a text box to a scroll bar in the Properties window.

TIP 10

The Math object methods sin and cos enable movement in a circle.

TIP 15

Use code to delete elements from, and add elements to, an Array.

TIP 16

Multiple elements can be added to create multidimensional Arrays.

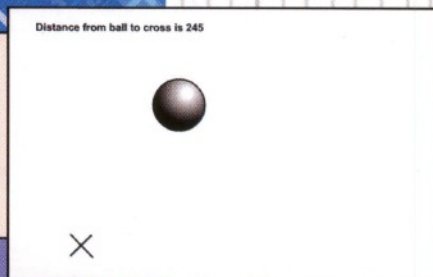
TIP 17

Arrays can be joined together to create a String - that is, a line of text.

60 ACTIONSCRIPT TIPS

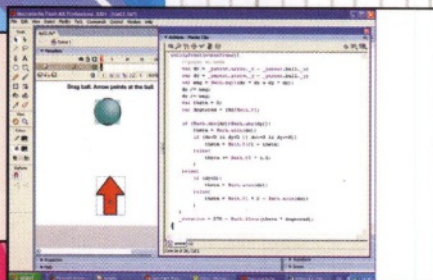
TIP 1

Work out the distance between two clips with some basic calculations.



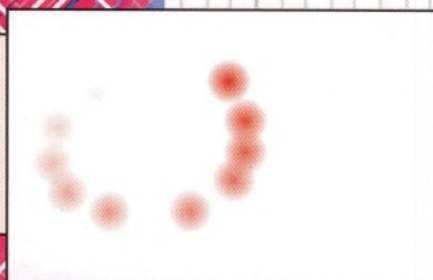
TIP 2

Determine the angle between the two clips before rotating the arrow.



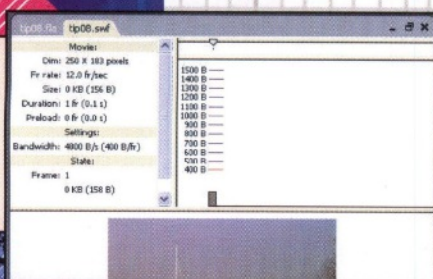
TIP 3

Create trails by using two MovieClips: individual and overall controller.



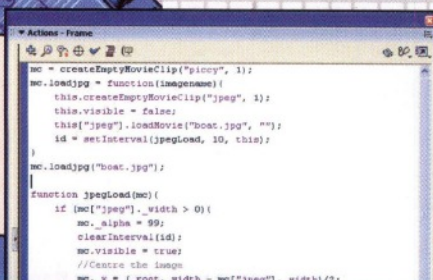
TIP 8

Create an empty MovieClip to make loading JPEG images easy.



TIP 9

A progress indicator enables you to see how long it takes JPEGs to load.



1. Calculating distances

You can calculate the distance between two MovieClips.

```
var dx = _parent.ball._x - _parent.cross._x;
var dy = _parent.ball._y - _parent.cross._y;
var dist = Math.sqrt(dx * dx + dy * dy);
```

Use the square root of the sum of the square of the distance apart in the x and the square of the distance apart in the y.

2. Point one clip at another

This tip first creates two variables, dx and dy, which are the distances apart of the two clips in the x and y axes. Then you get the distance between these points. Next, divide dx and dy by this distance to get what is called a unit length vector. You can use this vector to determine the angle between the two clips and hence rotate the arrow clip to point at the ball.

3. Creating trails

Use two MovieClips: one clip is the individual part of the trail and the second is the overall controller. In this example, the individual part is set to fade out over time using the `_alpha` property. The controller creates new clips, starting at level 1, on the fly, at the existing mouse position. Once the limit of clips is reached, the controller loops back to level 1.

4. Using loadVariables

The above code segment loads the contents of the file 'tip04.txt' from the cover disc into named variables at the root of the Movie. tip04.txt is simply:

```
&count=100&
&loaded=true
```

Each variable name/value pair is separated by an ampersand.

5. Using the LoadVars object

```
lc = new LoadVars();
lc.onLoad = function(success){
    trace(success + " " + this.count);
}
```

```
lc.load("tip05.txt");
```

The LoadVars object is a more elegant way of doing the same thing as in the previous tip. The principal benefit is you can define an onLoad function that's called when the data is loaded. To start the load, call the load method with a single parameter, the file to load.

6. Using the name Array

Every MovieClip has an array of names that can be used to access variables and Objects.

```
//Access any variable or Object within the MovieClip
this[myName];
//Access any variable or Object within the MovieClip's parent
this._parent[myName];
//Access any variable or Object within the root timeline.
_root[myName];
```

7. Using setInterval

To define something that occurs at a certain time interval or a timeout event, use setInterval. You pass a function that is called when the number of milliseconds in the second parameter elapses.

```
var id = setInterval(timeOut, 1000);
```

Use the return value from setInterval to clear the event using:

```
clearInterval(id);
```

8. Loading a JPEG image

Just create an empty MovieClip and load the image using the code below:

```
mc = createEmptyMovieClip("piccy", 1);
mc.loadjpg = function(imageName){
    loadMovie("boat.jpg", "");
}
mc.loadjpg("boat.jpg");
```

9. Adding a progress indicator

Use the following code to add a progress indicator to the JPEG Loader.

```
mc = createEmptyMovieClip("piccy", 1);
mc.loadjpg = function(imageName){
    this.createEmptyMovieClip("jpeg", 1);
    this.visible = false;
    this["jpeg"].loadMovie("boat.jpg", "");
    id = setInterval(jpegLoad, 10, this);
}
mc.loadjpg("boat.jpg");
```

The trick here is to create an empty MovieClip, called 'jpeg' within the MovieClip mc. Set the MovieClip mc to invisible and then load the jpeg into the sub clip. Set an interval caller so that you can use this to update the progress. Look at the code on the cover disc for details.

INTERMEDIATE

10. Opening a file requester

This code opens a file requester:

```
os = $version.substr(0, 1);
if (os == "W"){
    getUrl(".");
}else{
    fscommand("exec", "browseCD");
}
```

11. Repeating sections of code

The onEnterFrame function is good for code that needs to repeat at the pace of the movie's fps. Create it dynamically using:

```
this.onEnterFrame = function(){
    //Things to do each frame
}
```

12. Starting with the drawing API

The code below creates a MovieClip and then uses the drawing API to draw a black line that's two pixels thick from the point (275, 200) to the current mouse position.

```
this.onEnterFrame = function(){
    removeMovieClip("mc");
    mc = this.createEmptyMovieClip(
    "mc", 0);
    mc.lineStyle(2, 0x000000,
    100);
    mc.moveTo(275, 200);
    mc.lineTo( root._xmouse,
    root._ymouse);
}
```

13. Draw rectangle function

Flash has the ability to extend the functionality of the built-in objects. You can add a prototype, in this case a function to draw a rectangle, to the MovieClip object and then every MovieClip has the additional function, drawRect.

```
MovieClip.prototype.drawRect =
function(l, t, w, h, scol, fcol){
    this.lineStyle(1, scol, 100);
    this.beginFill(fcol, 100);
    this.moveTo(1, t);
    this.lineTo(1 + w, t);
    this.lineTo(1 + w, t + h);
    this.lineTo(1, t + h);
    this.lineTo(1, t);
    this.endFill();
}
```

14. Drawing a curve

Set the starting position of a curve using the drawing API method moveTo. Then determine two points: the end point and a control point that determines the bend.

```
mc = createEmptyMovieClip(
    "curves", 1);
```

```
mc.lineStyle(2, 0xFF0000,
100);
mc.moveTo(10, 200);
mc.curveTo(275, 0, 540, 200);
```

15. Drawing a circle

Use the drawing API method, curveTo, seen in the previous tip. Build a circle in 45-degree sections. Setting the end points uses the same method as beginners tip 10. The control points are halfway between the start and end points.

16. Drawing a pie chart

Extend the drawCircle method to create drawCircleSegment. Then generate a batch of data and use this to draw a pie chart. The code for this tip is on the cover disc.

17. Embedding fonts

Create a dynamic text box at design time and set this up to include the font outlines needed for the dynamically created text fields. Once the movie has the outlines required, set up a text field using createTextField to use the outlines just by setting the value for embed fonts to true.

18. Dynamic masking

Here, we create an empty clip and draw a rectangle in it. Then we create a second empty clip setting the first clip as a mask. By setting the mask as the drag item and loading an image into the second clip, we can move a window over the image.

```
mask = createEmptyMovieClip("box", 2);
mask.drawRect(0, 0, 100, 100,
0xFF0000, 0x00FF00);
mc = createEmptyMovieClip("piccy", 1);
mc.setMask(mask);
```

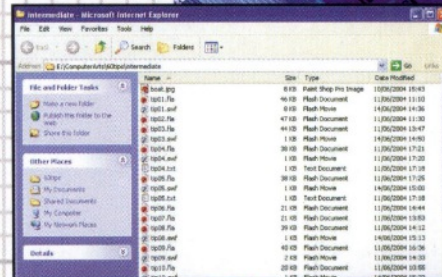
19. Using #include

You can write ActionScript in a text editor or use Flash MX 2004 Pro to add an ActionScript file (.as). To use the ActionScript in your movies, use this:

```
#include "tip19.as"
```

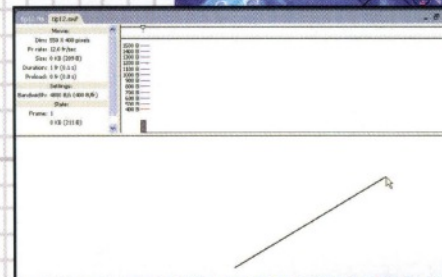
20. Email validation

The ActionScript File 'tip20.as' on the cover disc contains the code to validate an email address. Using simple String parsing, it checks if the '@' and '.' symbols are present by ensuring that there's a name before the '@' symbol and a domain after it. The function getValidationCode returns a value showing any errors that may have occurred.



TIP 10

Enable users to browse files using this handy piece of code.



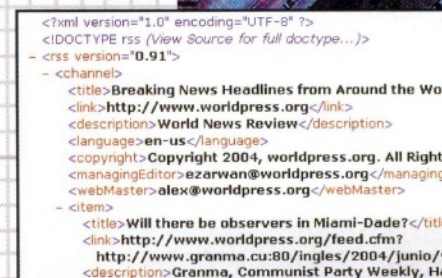
TIP 12

Dynamically draw shapes using the Flash ActionScript drawing API.



TIP 14

Draw a curve using the drawing API method moveTo.



TIP 16

Create a pie chart including a colour guide and text fields.



TIP 18

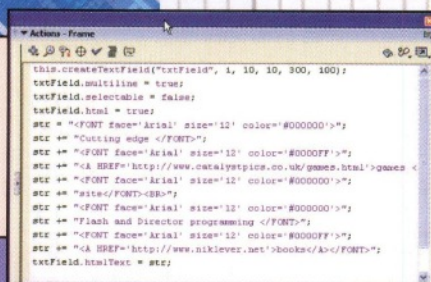
Dynamically mask an image by drawing a shape in an empty clip.



60 ACTIONSCRIPT TIPS

TIP 2

You can add HTML formatted text by adding the text field to an HTML field.



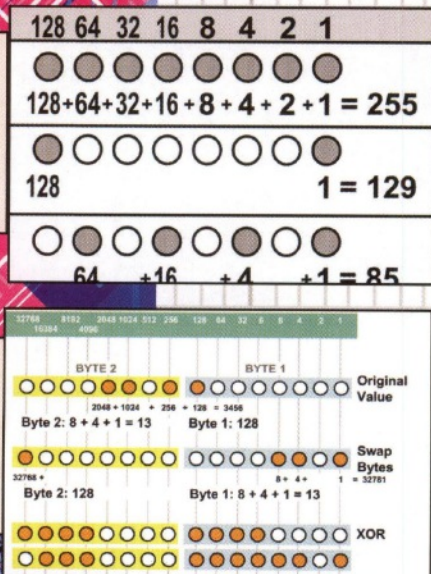
TIP 3

Creating a ContextMenu object enables you to add items to the menu.



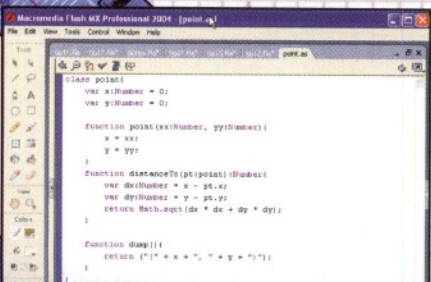
TIP 5

Manipulate data at bit level using bitwise operators.



TIP 7

Bitwise operators can be used to code info you pass over the web.



TIP 12

Get wise to using classes within an ActionScript file.

1. Additional TextField formatting

There are many properties of a text field created with createTextField. You can add a border using:

```
txtField.border = true;
```

You can size the border to fit the current text using:

```
txtField.autoSize = true;
```

You can set its type to an Input Field using:

```
txtField.type = "input";
```

2. Using HTML text

By setting the text field to be an HTML field, you can add HTML formatted text. In this example, we add two lines each with formatted text and a Hyperlink. The txtField has selectable set to false to make sure the correct cursor is displayed, and it has multiline set to true.

3. Altering the Context menu

To add items to the Context menu, you first need to create a custom ContextMenu object. Then you use the customItems Array to add content to the menu:

```
myArray.push(myContextMenuItem);
```

A ContextMenuItem is created using two parameters — the text to display and function to call. Finally, you set the context menu as the menu property of an Object on screen, in this case the _root Object.

4. Using multiple Context menus

You can attach a ContextMenu object to a specific button, MovieClip, or text field object, or to the entire movie. Use the menu property of the Button, MovieClip, TextField or _root classes to do this.

5. Using bitwise operators

At the most basic level, the data stored on a computer breaks down to either 1 or 0. This is a bit. You can manipulate data at the bit level using the bitwise operators.

```
n <<= 1;
```

The above code has the effect of moving the bits in n one to the left, which is the same as doubling its value.

6. Bitwise and, or and xor

For these operations, the bits in two variables are compared individually: And (&) — if both bits are on, the resulting bit is set to on. Or (|) — if either bit is on, the resulting bit is set to on. Xor (^) — if either bit is on but not both, the resulting bit is set to on.

7. Encoding and decoding

Using the bitwise operators it's possible to encode information that you pass across the web. For example, if a variable can only have values up to 65535 then it can be stored in two bytes. Get the first byte and the second byte and then swap them round. Xor the result and you have a garbled value. To restore it, do the operation backwards.

8. Using Listeners

Assuming that txtField is an Input Text Field then changing the content will have the result of calling the onChange method of the Listener Object.

```
myListener = new Object();
myListener.onChange =
function() {
    trace("Contents changed");
}
txtField.
addListener(myListener);
```

9. Responding to code events

When the text of a TextField is set in code, the onChange event is not triggered. To get around this, you can add a setText method to the TextField Object.

```
TextField.prototype.setText =
function(str) {
    this.text = str;
    this.broadcastMessage("onChanged");
};
```

10. Creating a point class

Classes are an excellent way of keeping your code well structured. They are easily created by writing a constructor function:

```
function point() {
    this.x = 0;
    this.y = 0;
}
```

To create a point simply call:

```
pt = new point();
```

The object will be created with the variables x and y set to zero.

11. Adding methods

You can add methods to the point class by using the prototype keyword followed by a method name.

```
point.prototype.init =
function (x, y) {
    this.x = x;
    this.y = y;
};
```


ADVANCED

12. Using a class file

An alternative approach is to create an ActionScript file and then create the class in the file. You declare any member variables and write a constructor function that has the same name as the class.

```
class point{
    var x:Number = 0;
    var y:Number = 0;

    function point(xx:Number,
yy:Number){
        x = xx;
        y = yy;
    }
}
```

13. Externally declared classes

To use an externally declared class, the class file needs to be in the same folder as the project file.

```
v1 = new vector(0, 1);
v2 = new vector(1, 1);
trace("Angle between " +
v1.dump() + " and " +
v2.dump() + " is " +
v1.angleBetween(v2) + "
degrees");
```

14. Making class variables private

The keyword 'private' means that the direct value of a variable in the class is not exposed. You can write methods to get and set them. It makes the code more robust.

```
class triangle{
    private var pt1:point;
    private var pt2:point;
    private var pt3:point;
    private var triarea:Number =
0;
    ...
}
```

15. Using simple physics

The bat moves and rotates using basic physics calculations. First, create a point 100 pixels up from the bat's centre point. Then calculate a force based on this position and the current mouse position. This is used to move the bat. Finally, we calculate a rotational force based on the current movement force.

16. Starting with XML parsing

The XML object is useful for accessing xml feeds from the web. While powerful, it can be a little confusing. The purpose of this small example is to generate a list of

data from any XML feed so that you can see the names and values of the many nodes. This takes a feed from a free news supplier.

17. Generating HTML from an XML feed

Once you've examined the feed, you can extend the parsing to generate anything your movie requires. In this example, we convert the news feed into HTML with hyperlinks. Only the nodes that have the nodeName, item, have the actual data required.

18. Using the LocalConnection object

The LocalConnection object is useful for communicating between two swfs running on the same PC. In this example, we create a receiving movie (tip18a.fla) and a sending movie (tip18b.fla). The sender communicates with the receiver using the connect name; it can then call methods on the receiver.

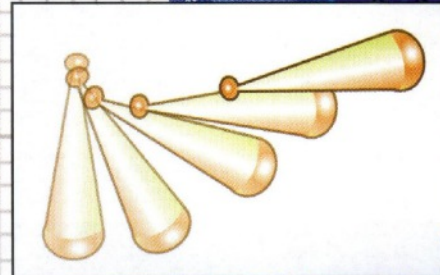
19. Using the SharedObject

The SharedObject is a good alternative to using cookies. You can store data on the client's machine or the remote server that can be accessed later.

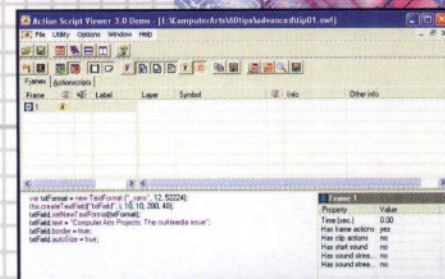
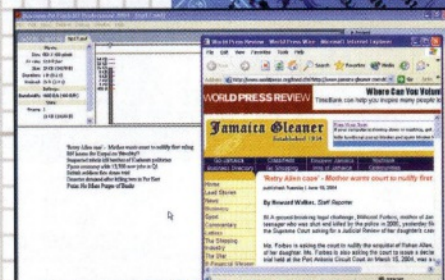
```
so = SharedObject.
getLocal("myInfo");
if (so.data.favoriteColor==unde
efined){
    //No saved data
    so.data.favoriteColor =
"orange";
    colors = new Array("red",
"blue", "green", "blue");
    so.data.possibleColors =
colors;
}else{
    //Show the saved data
    trace(so.data.
favoriteColor);
    trace(so.data.
possibleColors);
}
```

20. Using the ASV application

If you were to lose your project but still had the swf, you may well be able to recover the code using the ActionScript Viewer, which is a terrific application. The most current demo version of ASV can always be found at <http://www.swf2fla.com/asvdemo.zip>.



```
<?xml version="1.0" encoding="UTF-8" ?>
<DOCUMENT root="http://www.worldpress.org/">
<channel>
<title>Breaking News Headlines from Around the World, Powered by Wordpress.org</title>
<link>http://www.worldpress.org/</link>
<description>World News Review</description>
<language>en-us</language>
<copyright>Copyright 2004, worldpress.org. All Rights Reserved</copyright>
<manager>eznews@worldpress.org</manager>
<webmaster>eznews@worldpress.org</webmaster>
<item>
<title>Will there be observers in Miami-Sade</title>
<link>http://www.worldpress.org/feed.cfm?http://www.grenada.com/2004/jan/11/251est.html</link>
<description>Grenada, Communist Party Weekly, Havana, Cuba</description>
</item>
<item>
<title>Budget of W195.3 Trillion Requested for 2005</title>
<link>http://www.worldpress.org/feed.cfm?http://www.grenada.com/2004/jan/11/251est.html</link>
<description>Grenada, Communist Party Weekly, Havana, Cuba</description>
</item>
<item>
<title>Grenada backs BN on ISA</title>
<link>http://www.worldpress.org/feed.cfm?http://www.grenada.com/2004/jan/11/251est.html</link>
<description>Malaysia</description>
</item>
```



TIP 15

Move and rotate an object using simple physics calculations.

TIP 16

Access XML feeds from the web using the XML object...

TIP 17

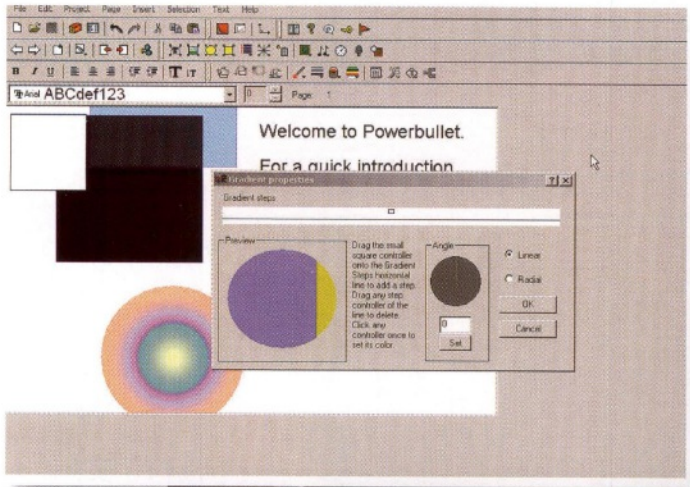
...and then use the feed to generate anything your movie requires.

TIP 18

Communicate between two SWF movies on the same machine.

TIP 20

The ActionScript Viewer can recover code from lost projects.



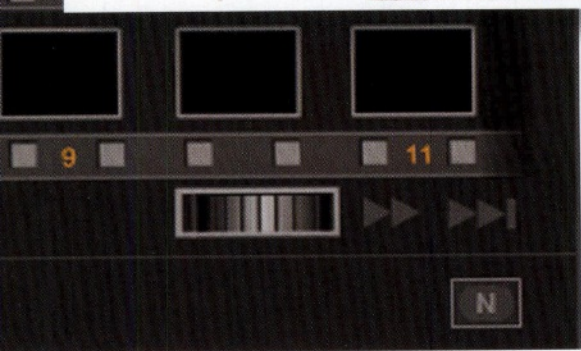
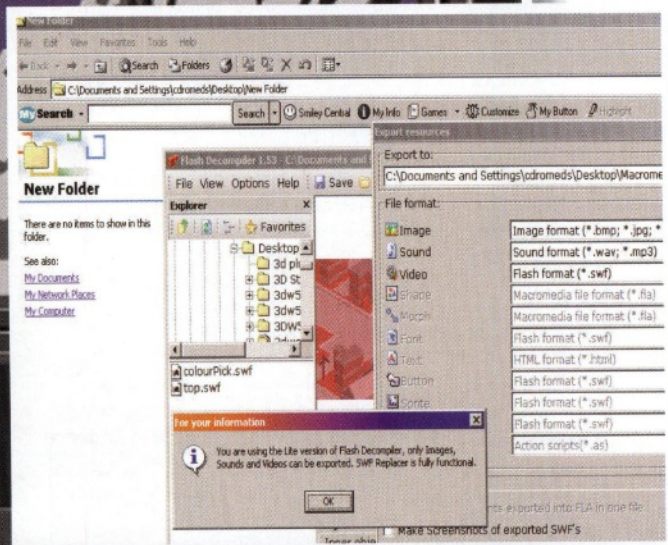
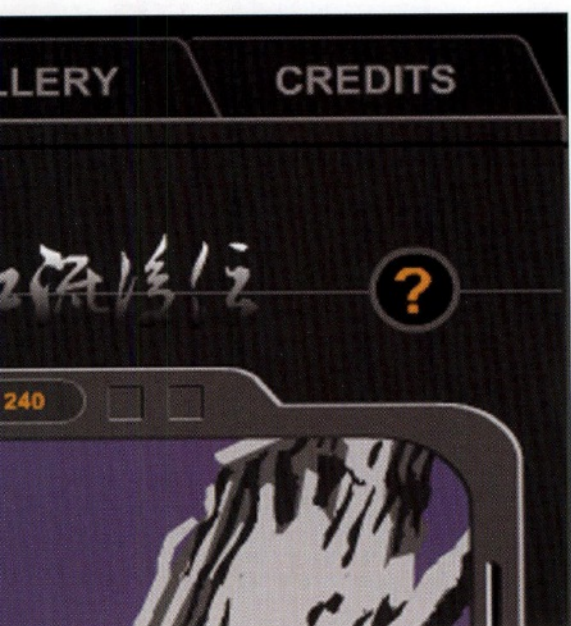
ON YOUR CD

5

Disc pages

A selection of the applications that accompany Macromedia Studio MX 2004 on your cover disc...

212	Flash Decompiler Lite
214	Flash Designer
216	Powerbullet Presenter
218	Comixware 2.1 trial: storyboard a short movie



ON YOUR CD

FULL PROGRAM

Flash Decompiler Lite application

Extract media from inside your compiled SWF files

PRODUCT INFO

PLATFORM
PC only

COMPANY
Eltima Software

WEB
<http://www.eltima.com>
(<http://www.decompiler-swf.com>)

PRICE FOR NON-LITE VERSION
€46.95

REGISTRATION NAME
Future Publishing

REGISTRATION CODE
000K3X-G99EX9-QHMEZ8-69C2Y1-B1ENAV-R5J4PD-A2N8T9-1308A1-06454C-A32E2D-49F751-CCC98D-7D2E8A

USEFUL HINT
You can simply copy and paste this code from the text file on the disc

Flash Decompiler is a useful application from Eltima Software for decompiling SWF files. It enables PC users to extract sounds, images, videos, shapes, frames, morphs, fonts, texts, buttons, sprites and ActionScripts from any SWF file (this Lite version is restricted to images, sound and video extraction, SWF Replacer is fully functional). All you'll need to do is navigate to the desired SWF with a built-in File explorer and click it. Alternatively, you can select the extraction option in right-click menu on any SWF file in your system directly. Decompiled resources are combined into groups for better convenience — to save them, you need to select necessary objects or tags and choose 'export' with another simple click.

There are a wide range of export options once you have extracted your SWF content. For example you can export sounds into WAV, FLA and MP3 formats, images into PNG, JPEG and BMP. Export options for video are currently only in FLV (which is a great format for online movies; however if for some reason you want the file in another format you need only convert). When extracting text you have options to export into RTF, TXT, and HTML formats. It can also handle SWF files where text can only be identified as symbols. The software will let you choose the corresponding

letters and, as the result, you will get readable text in the decompiled file. Flash Decompiler currently exports shapes, images, morphs, texts and sounds into FLA format for easier reconstruction of lost original Flash files. Future plans include full conversion of SWF into FLA and batch decompilation. Export formats are regularly updated so you can always contact the team at Eltima and suggest new ones yourself if your preferred format is not supported.

Edit without recompiling

Flash Decompiler also features SWF Replacer. This function lets you easily replace images, colors, lines and gradients in your SWF files; this approach completely changes the way SWF files are created. Using Flash Decompiler you can edit them yourself without the need to recompile into FLA. Plans are also afoot to incorporate sound, ActionScripts and Texts replacement. Flash Decompiler provides advanced decompilation of ActionScripts, making it possible to review the code of a SWF file that catches your attention. ActionScripts decompilation supports more than 50 tags combinations and incorporates a handy Search function through all the ActionScripts in a SWF file. In addition, ActionScripts can be viewed as Actions, pi-code, dump or pi-code+dump.

Other features include a simple way to browse your hard-drive for Flash movies, you can add files to quickly accessible Favorites folder and play selected items or preview decompiled SWF elements in a built-in Flash viewer. This Flash viewer enables you to drag the image, zoom in/out or view the original size at will. A range of Interface skins and layout options mean that you can customize Flash Decompiler to your personal taste, and the use of convenient hotkeys makes it easy to master its numerous functions and enjoy on-the-fly decompilation.

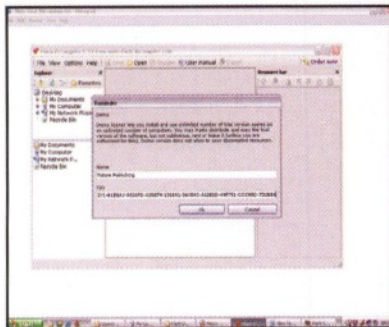
Flash Decompiler fully supports all Macromedia Flash versions and shows stable operation in different environments.

REGISTRATION INFO:

Registration name: Future Publishing
Registration code:
000K3X-G99EX9-QHMEZ8-69C2Y1-B1ENAV-R5J4PD-A2N8T9-1308A1-06454C-A32E2D-49F751-CCC98D-7D2E8A

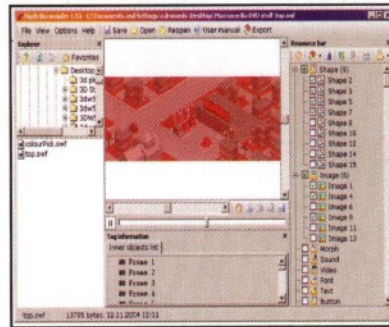
Full-featured registered version can be purchased from Eltima (www.eltima.com) for only €46.95

If you need extra hints and tips about using this program, the official website is a good place to start.



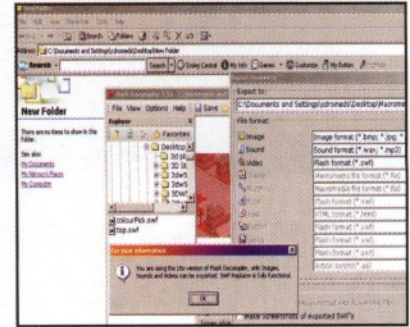
1 INSTALLATION

From the DVD interface simply browse to the Full Products section, select Flash Decompiler Lite and click the Install link. On initial installation, the program will be running in Demo mode — you will need to go to the Help menu, then select Change License type. Now just enter the user name 'Future Publishing' and the registration code '000K3X-G99EX9-QHMEZ8-69C2Y1-B1ENAV-R5J4PD-A2N8T9-1308A1-06454C-A32E2D-49F751-CCC98D-7D2E8A'. Don't worry about your typing skills: you can find this code in the text file in the Flash Decompiler folder on the disc.



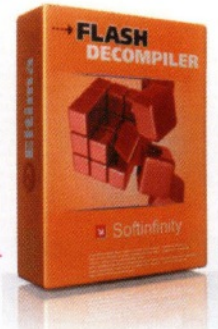
2 SELECTION

Use the easy to use browser to find the Flash movie you wish to decompile (we've selected a simple Flash animation from the Computer Arts Projects interface here in this example). Once selected, simply use the options to the right of the interface to select which elements you wish to extract. These elements can be images, movie files, text, sprites, etc.

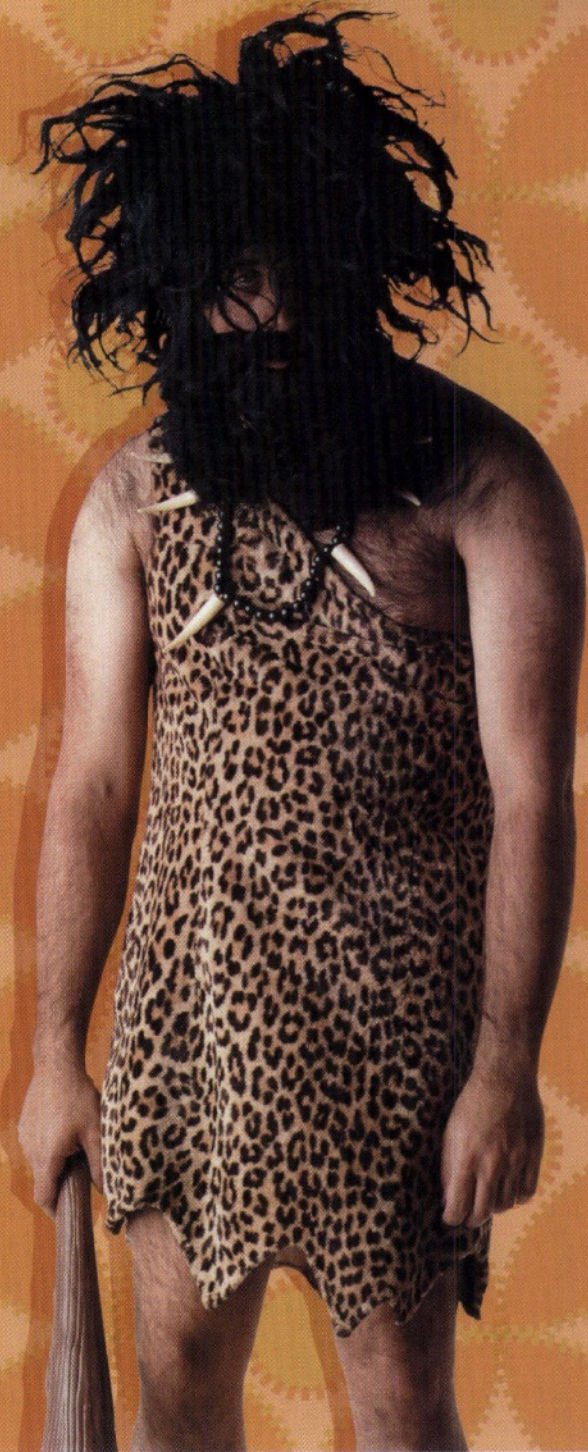


3 EXPORT

That is pretty much the job done. All you have to do now is choose which export options you wish to use and where on your system you want the software to export to. The output folder is logically split into categories for the different media types.



Move with
the times...
get your
favourite
magazines
online



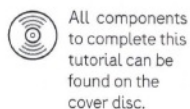
Make huge savings off the regular shop price
and choose your favourite magazine today!



myfavouritemagazines.co.uk

ON YOUR CD

Create Flash animations



All components to complete this tutorial can be found on the cover disc.

Liven up your web site with Flash animations the easy way by using the great facilities provided by Flash Designer

Flash can be great but creating animations for your web site can be tiresome and involve lots of time. What's more, if you go for Macromedia's Flash MX 2004 Professional, it can easily set you back nearly £600 — a lot of cash if all you want to do is to create simple animations for your home page. Step forward, Flash Designer, a little tool that makes it incredibly easy to build interactive Flash animations with minimum skills required. It not only simplifies the process, but speeds it up a great deal, too. You can design simple buttons, animated banners or interactive roll-over menus and upload them to your site in just a few clicks. Flash Designer allows you to animate images, shapes and texts and comes with more than 130 ready-made effects (including zooms, flips, squeezes and rotations). You can even add sound to your Flash movie.

The program is so easy to use because

you just need to import a graphic (you can use your own photos), select it and then choose either intro or outro animations for it. As Flash Designer contains various drawing tools (such as circles, polygons, rectangles), even graphics can be produced very quickly

In the Frame

Flash Designer works with frames which you can easily re-order if you want to change the look of your animation or vary the delay between each frame — a feature that comes in particularly handy for the creation of animated banners.

It's also possible to export animations in .swf format and generate the HTML code you need to integrate the Flash movie with your web page. The built-in browser display allows you to preview your animation exactly as it will look on the web. Finally, you can convert GIF animations to Flash animations.

Flash Designer is an incredibly versatile, powerful and, above all, useful tool. It works

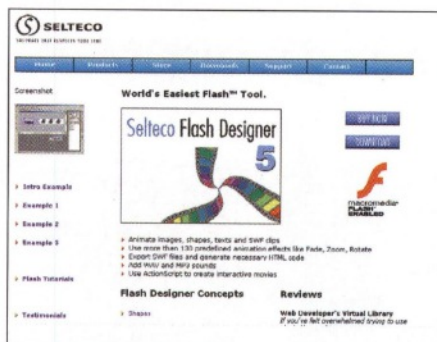
with all HTML editors including Dreamweaver and FrontPage. If you want to add Flash effects to your web site, this program is for you.

More information

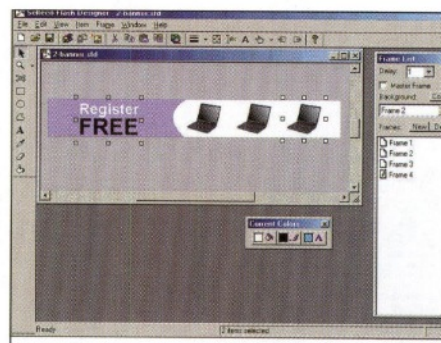
To find out more about Flash Designer, visit its site at www.selteco.com/flashdesigner, which includes explanations of the types of animation you can apply (eg sprites, gradients, and tween animations) plus a list of detailed tutorials (eg how to create scroll text). There's also a support section containing shorter tutorials. For example, you can learn how to use Flash Designer with PowerPoint, how to create an invisible hotspot and how to link an animation to a URL. In our tutorial, which will help you understand the basic functionality of Flash Designer, we'll show you how to complete one of the most popular Flash projects: making a simple animation with some text.

UPGRADE – More animations with version 5

If you enjoyed trying out the free version of Flash Designer, you could consider upgrading. The tool has developed a great deal and version 5 comes packed with many new features. For example, you can add sound in WAV and MP3 format and create animations in addition to intro and outro effects. The new version also allows you to insert existing SWF and Flash Designer clips and load and display SWF files dynamically to save space. This version is even simpler to use as it includes wizards to create slide-shows and animations in just a few clicks. If you'd like to upgrade to Version 5, visit www.selteco.com/store/netmagazineflash.htm, where you can purchase it for only £19.90 (reduced from £24.90).



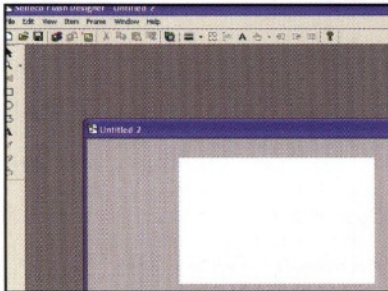
Flash Designer's home page is packed with more tutorials and examples of what you can achieve with this handy app



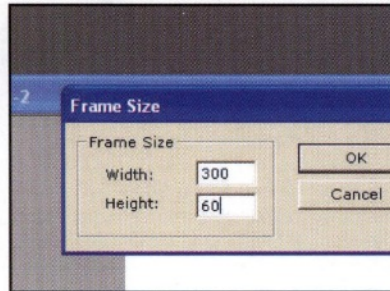
The latest version of Flash Designer makes it even easier to create Flash animations for your web site

Flash up your site

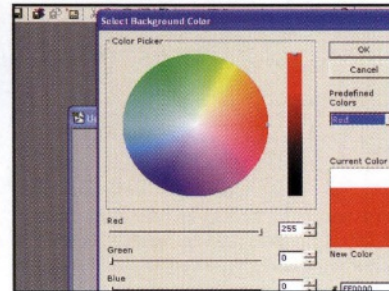
Getting started with Flash Designer



1 When you launch Flash Designer for the first time a welcome dialog will pop up. It lets you choose how you want to start (eg with a quick animation or with the last edited file) or whether you'd like to learn more about the tool first. Leave blank document ticked and click OK.



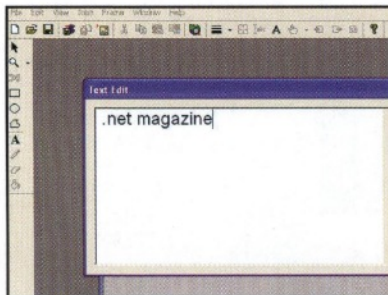
2 A blank document will appear in the main window. From the Frame menu choose Frame Size. Now you've got the opportunity to specify the size of your animation. By default the width will be set for 300 and the height for 200. Change the latter to 60, click OK and notice how your document has changed.



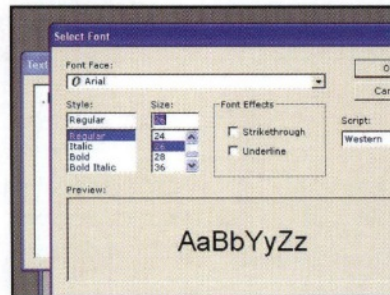
2 Go back to the Frame menu and choose Background colour. Here you can define the look of your animation by either picking a predefined colour or mixing your own with the colour picker and the red, green and blue sliders. When you've chosen a colour (we went for red), click OK.

More know-how!

The publishers of Flash Designer have a great selection of Flash tutorials online that perfectly complement their products. Visit www.selteco.com/tutorials/ for details.



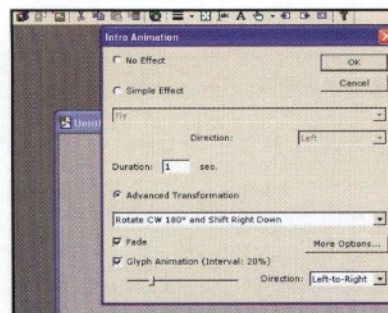
4 Now choose the text tool from the toolbox on the left and position the cursor over the red rectangle. When you click on it, a text edit dialog will appear. Enter some text in the box – later this will appear as part of your animation. You can also align the text and alter its colour.



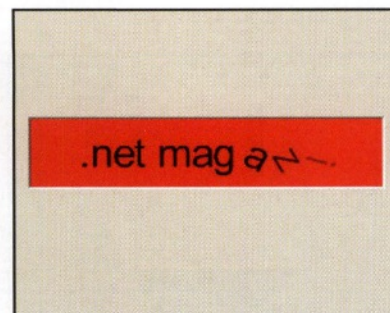
5 To change the font, click the Font button on the text edit dialog. Another window will pop up that allows you to set the font face, style, size, script and effects like strikethrough and underline. You can also preview the changes you're about to make. When you're done, click OK twice to close both dialogs.



6 Have a look how the text appears on your screen. It might not fit the rectangle properly, but you can move it around with the Select tool (from the toolbox on the left). Hold and drag the text into the correct position. If it still doesn't fit, try resizing your document or cut the text.



7 Go to Item > Intro Animation. Now you need to define whether you'd like to apply no effect, a simple effect or an advanced transformation. For our animation select the latter, choose Rotate CW 180 and Shift Right Down from the drop-down menu, tick Fade and Glyph Animation, move the slider to 20 and click OK.



8 Now go back to the Frame menu, choose New and observe the change in the frame list on the right-hand side. All you need to do now is to choose Play Animation from the Frame menu. If you're happy with your results, save the document by going to File > Export SWF File.



9 Go the File menu and select Export HTML Page. Your browser will display your animation at the top of the page and the HTML code you need to insert in your page copy underneath. Also make sure you copy your saved .swf to the same folder where your HTML page is located.

ON YOUR CD

The perfect pitch



All components to complete this tutorial are on the cover disc.

Create Flash presentations with ease using Powerbullet Presenter

Creating a presentation with real wow factor can be tricky these days. If you are going to slap a laptop down on a desk and make people sit through an hour of your sales patter, they'll expect some snazzy stuff and Powerpoint alone just isn't going to cut it.

So how are you going to keep people's ever-decreasing attention spans focused? The answer could be Powerbullet Presenter, a free program that creates presentations in Flash — great news for web designers. With

just a few clicks you'll be able to pep up your presentations with some great animations.

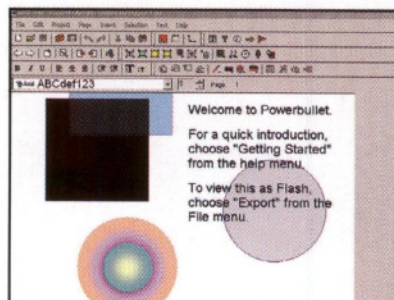
As long as you know how to confidently click your way around a typical word processing application, you'll find that Powerbullet Presenter's functions will seem pretty familiar. The program is specifically designed for creating multi-page presentations that are commonly used in brochures and catalogues, slideshows and projects and it's a great place to pick up a few general Flash tips, too.

The most commonly used operations are available with a button click or two, and there are handy keyboard shortcuts for most operations. For instance, using the up and down arrow keys — in conjunction with the Control and Shift key — increases or decreases the font size.

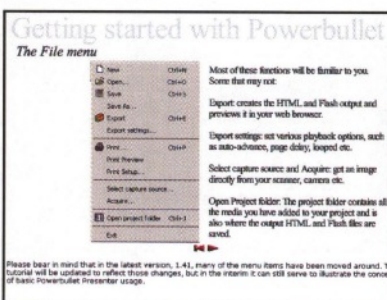
Powerbullet Plus is available for \$19.95 (£10.71) and offers a few advanced features such as rollover effects, custom navigation buttons, advanced click actions and timed animations.

Making your first presentation

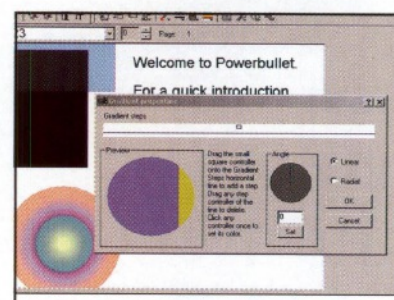
Amaze your boss by using something other than the ubiquitous MS PowerPoint...



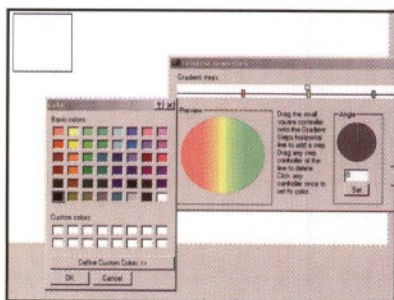
1 Launch Powerbullet for the first time and you're presented with a simple introduction screen with three toolbars; many of the icons will probably look familiar. There are plenty of new ones as well so, to get to grips with programs functions before you start making presentations, click Help then Getting Started.



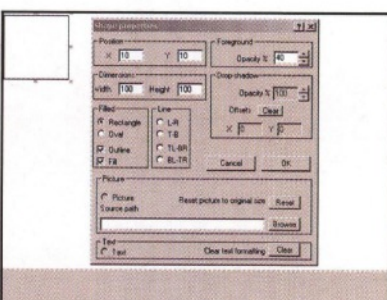
2 You will now be taken to a web-based tutorial in a new window. Here you can cycle through a series of introductory pages outlining how to use the program's basic functions. Click on the red arrow button to proceed through the web presentation and learn more about Powerbullet.



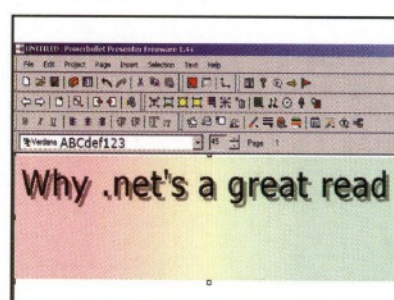
2 Once the tutorial is finished, you can get started with your new presentation. We're going to create a simple electronic brochure. To create a title, select Rectangle from the Insert menu on the toolbar. Then select Gradient Setup from the Selection menu — a window will appear showing the properties of your new rectangle.



4 At the top of the gradient box you'll see a small square. Left-click this and drag it to the line below it to begin adding colour to your title box. Repeat this action with two more boxes spacing them out evenly. Then click in each box and select a colour from the menu that will appear. Then click OK.



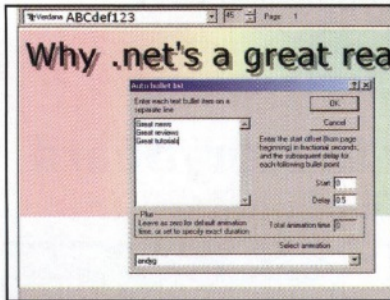
5 Click on the rectangle you created earlier so the grab handles are visible. Right-click on the rectangle, select Item Properties and change Foreground Opacity to 40%. Grab the middle right-hand grab handle and drag it to the right. You'll see the colours of the boxes you have chosen merging into one.



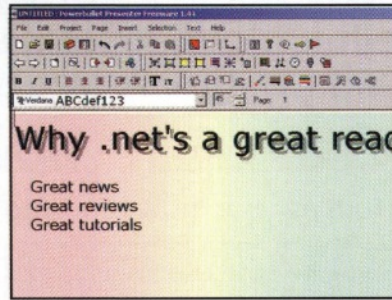
6 Now select Text from the Insert Menu. A small text box will appear, in which you can type a presentation title of your choice. Once done, click outside the box and then reselect it again. Right-click and select Item Properties again, change the Drop Shadow to 40% and enter an offset of six for both the X and Y axes.

Making your first presentation continued...

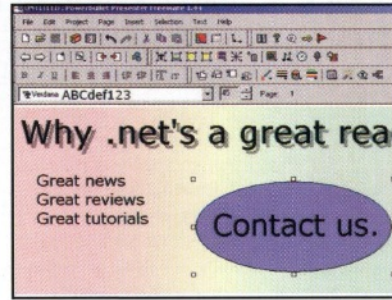
Remember to tell your appreciative audience where you got the program!



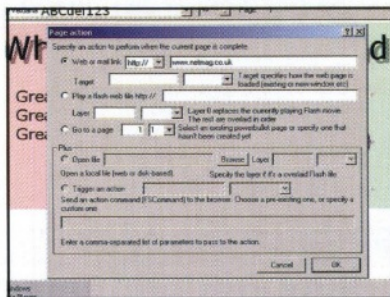
7 Next, add a list of animated points by going to the Insert menu again and clicking on Bullet List. Type your selection in the space provided and select 'Bump in from right' from the Select Animation box (there are dozens of animation choices, as you will see) then click OK. Now save your project by clicking on the disk icon..



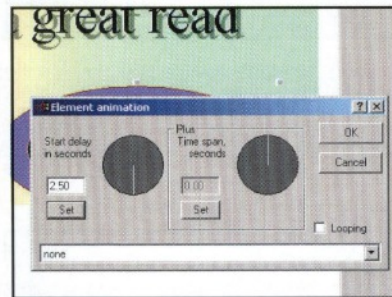
8 Now, from the Page menu click Preview, this will show the Flash animation you've created, and you'll see the first stage of your handiwork unfold. Close the browser window and return to the program. Now you can start developing the presentation. To add a Contacts box, select Oval from the Insert menu.



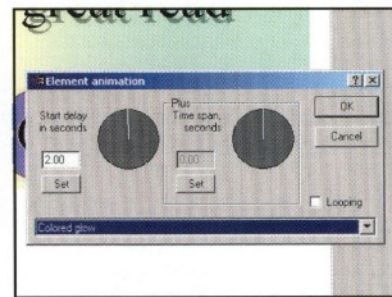
9 Click on the oval and pick Fill Colour from the Selection menu. You can also then select Line Colour (the colour of the outline). Then click outside the oval, create a text box by selecting Insert > Text. Type some text, Contact us, for example, then drag the oval over the text box and reshape it to your liking.



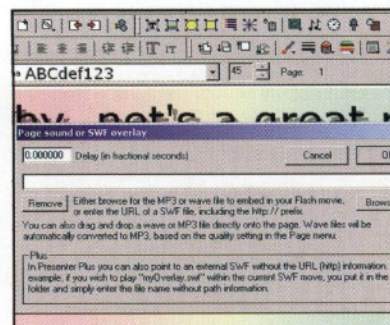
10 With the oval Contact us box selected, choose Page Link from the Page menu. There are a variety of web-based options here but, for now, just enter a URL in the top box to link to your web site. This link function specifies an action to be executed when the presentation pages finish.



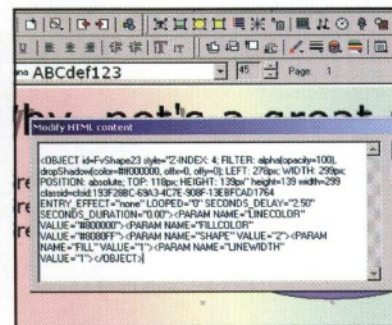
11 To start animating the Contact us text, select the oval, right-click on it and choose Item Animation from the menu that appears. Here we're adding a timer to delay the appearance of the text in the Contact us box. About two seconds will do, then set the time and then click OK.



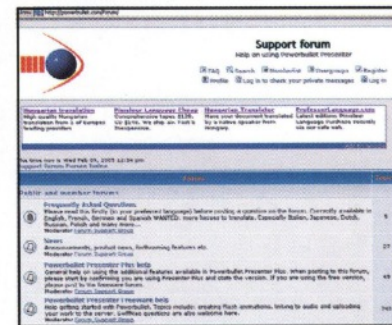
12 To make the text glow, select the Contact us box and choose Item Animation as well. Set this time to about two seconds and set the type of animation to Coloured Glow. Now your simple animation comes alive and you can start to use more expert tools to create something even better.



13 You can also add sound to your presentations. Select the Page menu and pick Page Sound. Either browse for your MP3 or WAV file to embed it into your Flash movie or enter a URL of the MP3 or SWF file. As with the animation, you can select a delay time for the sound effects.



14 For advanced users the first thing you'll want to know is how to add proper code. Simply click on the View and Edit HTML button and a window will appear containing the basic code. Experiment with the item's code to add professional touches to the basic Powerbullet functions.



15 After a successful introduction to Powerbullet you can start to branch out and the handy Powerbullet Support Forum (linked to from the Help menu) is a great place to pick up tips and tricks for producing better presentations. The FAQs and Bugs sections are the ones you'll want to keep a close eye on.

Keep up-to-date

Programs like this are constantly being updated so you'd be wise to click on the Flag icon on the Powerbullet toolbar. This will link you straight to the Powerbullet updates section on the web and you can update your program. <http://powerbullet.com>

ON YOUR CD

Comixware 2.1 Trial

Need to sell your idea visually? Whether it's a proposed intranet development at work, a 30-second commercial or music video, nothing is more persuasive than a dynamic storyboard geared with multimedia presentation and interactivity, as **Ting Li** explains

Prepare a panning element

There's a twist for preparing the panning foreground element (tuto_gz_fg_pan.swf), which starts off-stage and then moves halfway into the scene. If you use the same method as described in step 5, the thumbnail won't show up in the Asset Bin. To remedy this, add a `play()` command at the first frame of the Flash movie (tuto_gz_fg_pan-v2 fla). This will make the thumbnail play continuously in the Asset Bin so that you can grab it.

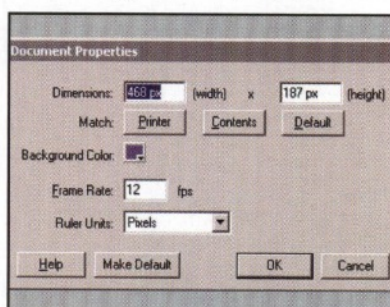
Professional motion graphic designers, screenwriters and film makers know that the best way to plan a multimedia project is by storyboarding. But a storyboard isn't just a great way of planning; it's also a powerful selling tool that can help convince clients of the power of your ideas in comprehensible visual terms. For more on storyboarding with Flash, see page 130. Comixware is a specially tailored multimedia storyboarding app that enables you and other creatives to compose a story dynamically and play it out instantly in a movie-style presentation. You can then add materials, make changes and fine-tune the project in real time – even remotely, courtesy of the software's built-in communication tools. This tutorial assumes you have a basic operating knowledge of Comixware. If not, it's worth checking out the "Introduction" chapter of the app's Help manual.

Assets of Fire

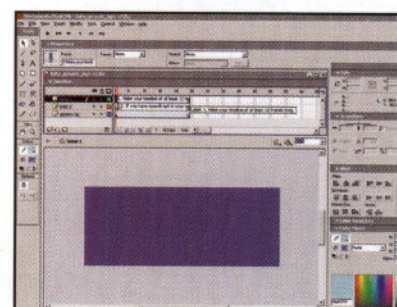
There are three types of assets used in Comixware for movie composition: Timeline, Scene and Music. You'll use Timeline to build the movie's timeline, Scene to lay elements on top, and Music to provide the soundtrack. Timeline assets can include a movie clip or static pictures, and serve as the "background" layer, while Scene assets are the elements that sit on top. Note that you can use video clips and pictures as Timeline or Scene assets. Here, you'll be shown how to prepare artwork and build a scene for a Comixware music video. The sample music video is called Wadidyusay, which has been sampled from Zap Mama's eponymous song (www.creativecommons.org/wired).

Comixware: how to storyboard

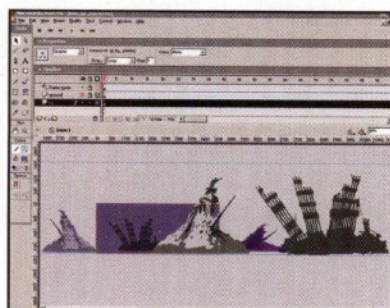
Turning your inspiration into a step-by-step format



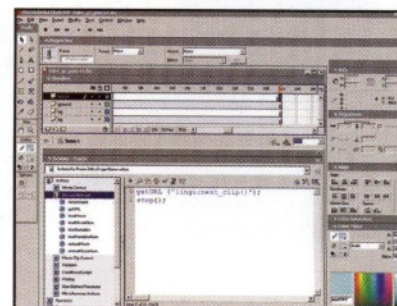
1 First, prepare the background clip – an animated panning background – for scene #2. Create a new file in Flash MX and name it tuto_gz_pan fla. Create the document with a dimension of 468x187 pixels (or a document of the same proportions).



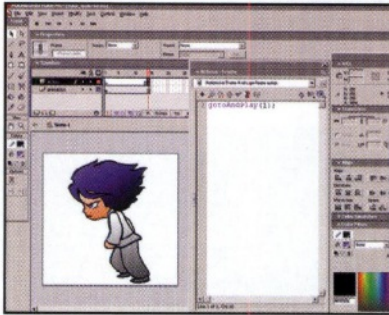
2 The background is an animated Movie Clip called gz_bg_panning at the root level of the Flash movie (tuto_gz_pan-v2 fla). Change the Behaviour of this instance from Movie Clip to Graphic, and select Loop from the graphic's Options menu. Make the root timeline length equal to the animation timeline.



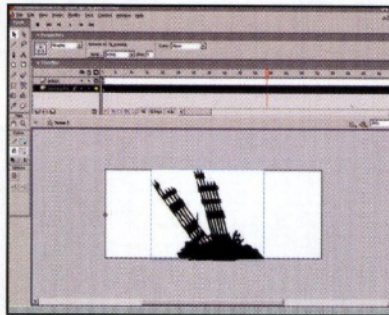
3 To prepare a one-way animation or static image clip you should place your artwork at the root level. See sample source file tuto_generic_bg1.v2 fla for further details.



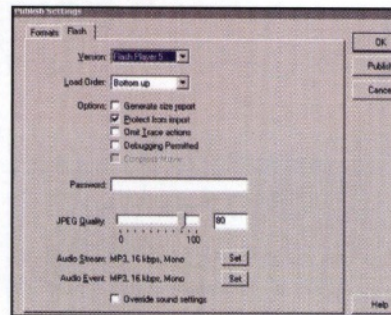
4 The preparation for Flash background clips in version 1.2 is slightly different than shown above. The most important difference is that you need to place two lines of ActionScript at the last frame of your movie. See sample source file tuto_gz_pan-v1 fla and tuto_generic_bg1.v1 fla for more information on how to do this.



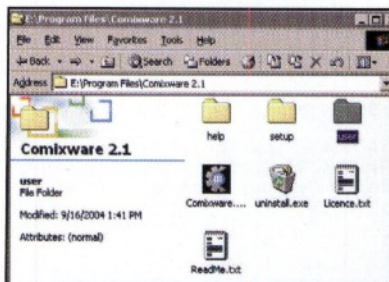
5 Now prepare a walking character for this scene. Creating an “animated” element is identical to creating an “animated” background clip – except that for a cycle animated element you need to add one line of ActionScript to the last frame of the movie (see tuto_walk-v2.fla).



6 For static prop elements, simply place the graphic at the first frame of the movie. Take a closer look at tuto_gz_fg_ground.fla to see how this is done.



7 Once you have all background clips and composition elements prepared, go to File and select the Publish Setting dialog. Select “Flash Player 5” from the Versions drop-down menu and publish all your swf movies in Flash 5 format.



8 Go to Comixware’s application directory and select the “user” folder on the cover CD. Inside you’ll find four sub-folders: “backgrounds”, “elements”, “music” and “save”. Place your assets into the respective folders: Timeline assets in “background”, Scene assets in “elements”, Music assets in “music”, and the sample movie file “Wadidyusay” in “save”.



9 Now launch Comixware and go to the Compose section. Click on the “bin” tab and go to tag #5. You should now see thumbnails of all your timeline assets.



10 Drag the thumbnail of that panning background clip from the Asset Bin to the timeline and drop it in scene #2. You’ll see the first frame of the clip will appear immediately in the Preview window.



11 Double-click scene #2 in the timeline to switch it to Composition mode. A larger Compose window will now replace the Preview window. The contents of the Asset Bin will also have changed to Scene assets. Click on tag #5 and to view all of your composition elements.



12 Now drag-and-drop the composition elements to the stage in the layer order that you’d like them to be composed. In this sample, we opted for: “ground”, “shadow”, “walking character”, the animated “leaves”, and lastly, in the foreground, “debris”.



ON YOUR CD

Comixware 2.1 Trial: how to storyboard continued

Getting your visual composition ideas across to others has never been easier

Scene length control

Scene length control is a useful and unique timing device that can not only modify the length of a scene, but also change the behaviour of the animation. You can continuously make length changes to a scene by re-entering new numbers in the Total Frame field. Each change is based on the last "modified" parameter. So, by combining different options, you can change one animation into something completely different.



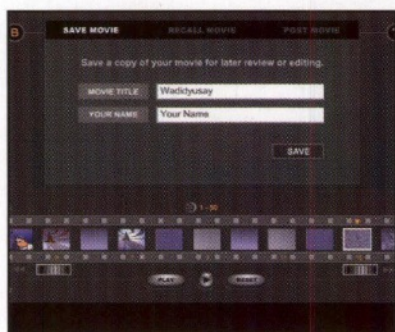
13 Use the bounding box controls that appear around each element to adjust the size, position, orientation, layer and transparency of each one. You can tug the Asset Bin into the Compose window if you need more room.



14 Use the frame-slider to navigate the frame-space within the scene. When an on-stage element is selected, the frame slider will turn into "in" and "out" marks. Drag these marks around to change the duration and the placement of the object within the scene's timeline.



15 Replace the existing total frame number (240) with the desired total frame number (84) and hit Return. Select the To End option from the Scene Length menu. Now 156 frames are trimmed from the end of the original clip, leaving the scene with a new length of 84 frames.



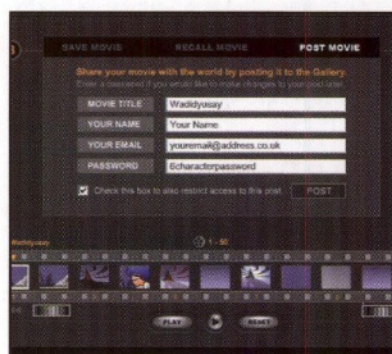
16 Once you're happy with your composition, press the (red) Record button to save the scene to the timeline. Click the Edit Timeline button to switch back to Timeline Editing mode and repeat the above process to build additional scenes for your timeline. Now save your movie.



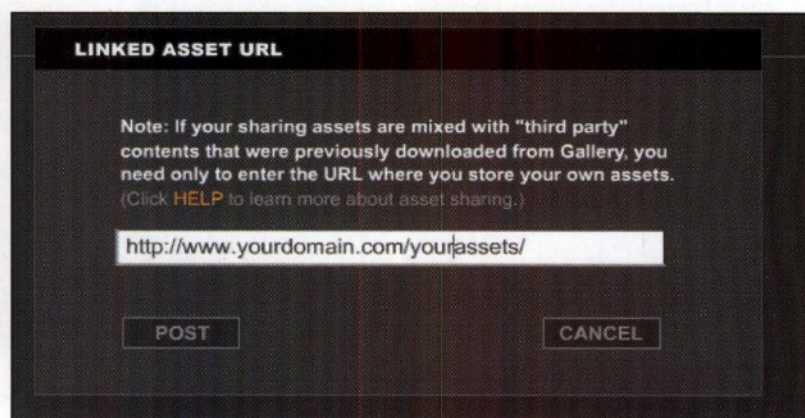
17 Double-click the Sound button on the right-hand side of the timeline to bring up the music panel. Click User and then select a soundtrack from the list for your movie.



18 Now press Play to view your movie in the Play section. Go back to the Compose section for additional editing, or to save the movie if you're happy with it.



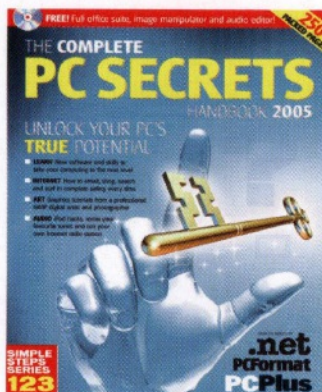
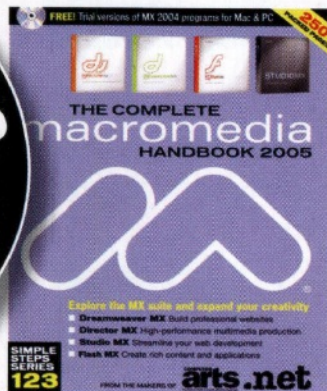
19 You can share your movie with remote users by posting it to Gallery. Just go to the Post Movie screen in the Save section and fill out the posting form. Choose a password for later editing options and check the Restrict Access box if you want your movie to be password protected.



18 If your movie contains "user" assets, you need to upload them to a designated internet server to give access to remote users. When you're asked to specify a URL for these linked assets, enter the web directory in which you stored them and hit the Post button.

COMPLETE EXPERTISE COMPLETE KNOW-HOW COMPLETE HANDBOOKS

SUBSCRIPTION OFFER



WHEN YOU SUBSCRIBE TO

THE COMPLETE HANDBOOK SERIES

***WHEN YOU PAY BY DIRECT DEBIT**

The Complete Handbook Series will bring you:

- All the information you need to improve your knowledge of the essential creative subjects
- Carefully chosen collections of the very best tutorials and practical projects from Future Digital's award-winning magazines
- Unbeatable coverage of each specific subject area, with over 250 pages written by industry experts, plus a free CD-ROM with supporting files and additional software
- Free delivery direct to your door, so you never miss an issue

YES! I would like to receive 6 issues of The Complete Handbook Series

☐ UK Direct Debit £16.86 – (saving 25%) every 3 months, please complete the form opposite

☐ UK Cheque/Credit Card £71.99 (saving 20%) ☐ Europe £75.00 ☐ ROW £80.00

YOUR DETAILS

Title Mr ☐ Mrs ☐ Ms ☐ First name _____ Surname _____

Address _____ Postcode _____ Country _____

Tel no. (inc STD) _____ Email address _____

We will use the contact details supplied to communicate with you about your Complete Handbooks subscription. Indicate here if you're happy for Complete Handbooks, Future Publishing and its sister companies to contact you by: ☐ email ☐ mobile.

If you're also happy for us to pass your details on to carefully selected companies indicate here: ☐ email ☐ mobile.

METHOD OF PAYMENT

I understand that I will receive 6 issues over the next 12 months

☐ Direct Debit (UK only) ☐ Cheque (£ sterling drawn on a UK bank account payable to The Complete Handbook Series)

☐ Switch ☐ Mastercard ☐ VISA ☐ American Express

Card no

Expiry date / Switch only: Valid date / Issue no

Signature _____

Send to: THE COMPLETE HANDBOOK SERIES

Future Publishing Ltd
FREEPOST RLSC-SXSE-SKKT

Unit 4

Tower House
Sovereign Park

Market Harborough

Leicestershire LE16 9EF

Order code M003 Offer ends: 18th July 2005

PLEASE COMPLETE THE FORM OPPOSITE

Please indicate if you Do Not wish to receive relevant information about special offers or products/services from Complete Handbooks, Future Publishing or any of its sister companies by ☐ post or ☐ telephone.

Please indicate here if you do NOT wish us to pass your details onto other carefully selected companies to enable them to contact you about their products/services by ☐ post or ☐ telephone.

For UK new orders & renewals - 0870 8374722

For UK queries - 0870 8374773

For overseas new orders & renewals - +44 (0) 1855 436794

For overseas queries - +44 (0) 1855 436795

Instruction to your bank or building society to pay direct debit



Originator's identification number

7 6 8 1 9 5

1. Name and full postal address of your bank or building society branch

To: The Manager

bank/building society

Address _____

Postcode _____

2. Name(s) of account holder(s)

3. Branch sort code (from the top right-hand corner of your cheque)

4. Bank or building society account number

5. Instruction to your bank or building society

Please pay Future Publishing Direct Debits from the account detailed on this Instruction subject to the safeguards assured by the Direct Debit Guarantee. I understand that this instruction may remain with Future Publishing and, if so, details will be passed electronically to my bank or building society

Signature(s) _____ Date _____

Ref No (Office use only) _____

Banks and building societies may not accept Direct Debit Instructions on some types of account

EACH MONTH COMPUTER ARTS PROVIDES YOU WITH...

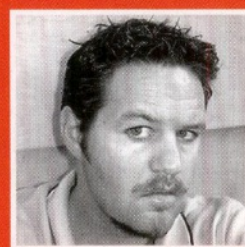
- A free cover CD, complete with full software, trials and essential utilities
- The chance to learn new techniques from a diverse range of cutting-edge tutorials
- All the latest news and impartial reviews from our team of industry experts



"Where else can I learn about the ultra-cool 123Klan, read about the legendary David Carson, or be

inspired by an unknown young designer? Forget the other magazines, *Computer Arts* has it all!"

Rob Chiu, motion graphics designer
www.theronin.co.uk



"Most design publications have become so distant from the heart of the design community. *Computer Arts* is more like a good friend's portfolio than some austere glass-tower publication."

Todd Purgason, creative director
<http://juxtinteractive.com>

**BUT DON'T
JUST TAKE
OUR WORD
FOR IT**

SUBSCRIPTION ORDER FORM

UK £54 (rrp £78)

Your subscription will start with the next available issue

YOUR DETAILS

Title	First name
Surname	Company
Address	
Postcode	
Tel no. (inc. code)	home/work
Email address	

PAYMENT DETAILS: I understand that I will receive 13 issues over the next 12 months

☐ Cheque (Made payable to Computer Arts. Sterling cheques drawn on a UK bank account)

☐ Mastercard ☐ Visa ☐ American Express ☐ Switch

Card no.

Expiry date Issue number

Signature

Date

We will use the contact details supplied by you to contact you about your subscription. Indicate here if you're happy for *Computer Arts*, Future Publishing and its sister companies to contact you with other information/offers by: ☐ email or ☐ telephone. Please indicate here if you do NOT wish us to pass your details on to carefully selected companies, to enable us to contact you about their products/services by ☐ email or ☐ telephone.

Return address Computer Arts Subscriptions, Future Publishing, FREEPOST RLSC-SXSE-SKKT, Unit 4 Tower House, Sovereign Park, Market Harborough, Leicester LE16 9EF

THREE WAYS TO ORDER

1 Online

Visit our online subscriptions service at:
www.myfavouritemagazines.co.uk/coa/m008

2 By post

Complete the form provided and return it to:
**Computer Arts Subscriptions,
Future Publishing,
FREEPOST RLSC-SXSE-SKKT,
Unit 4 Tower House,
Sovereign Park, Market Harborough,
Leicester LE16 9EF**

3 By phone

Call **0870 837 4722** quoting ref M008

Order code: **M008**

Offer ends: **18 July 2005**

SUBSCRIBE TODAY AND SAVE 30%

Now you need never miss an issue again...



PAY JUST
£54 FOR 13
ISSUES
(RRP £78)

WHY SUBSCRIBE?

- Subscribe before 18 July and save 30 per cent
- Get every copy delivered direct to your door
- Exclusive subscriber copies come with a stylish text-free cover
- Ensure your copy of the best international creative resource

To take advantage of this great offer visit

www.myfavouritemagazines.co.uk/coa/m008

SUBSCRIBE NOW
AND GET ACCESS TO
HUNDREDS OF FREE
TUTORIALS ONLINE





Index

1,2,3...

1st Page 2000	16
2D images	160
3D	40, 79
3D animation	130
3D drop-downs	145
3D Studio	37
3D to 2D conversion	79
3ds Max	40, 145

a

accessibility	24, 36, 51-59
actions	165
ActionScript	33, 70, 78, 73, 75, 80, 93, 97, 101, 123, 126, 128, 150, 156, 166, 170, 176, 192, 202-209
Adobe Photoshop	see Photoshop
Adobe Premiere	135
Advertising production	41
AfterEffects	130, 143
Alert component	192
aligning objects	190, 192
alistapart.com	92
Alpha transparency	181
Alt tags	52
animated type	143
animation	72, 78, 145
animation, characters	152, 156, 160
animation, GIF	186, 193
animation, looped	119
anti-aliasing	182
array	202
ASP	94, 96, 97
Audio	161
AutoCorrect	17

b

Backup!	43
bandwidth	35

batch process	185
behaviours	194
Berners-Lee, Tim	174
black and white	196
BMP	185
Boardmaster	131
broadband	32, 37
Bullet points	188
buttons	100, 127, 150, 185, 202

C

CERN	174
character animation	152, 156, 160
clear	87
clients	43-45, 133
Clusta Design Agency	143
CMS	102
CMYK	169
colour matching	185
colours, web-safe	180
Comixware	219
component design	121
construction process	30
Contribute	102
copyright	107
Corbis	38
countdown clock	164
Crash.net	11
cross-platform	90
CSS	53, 176, 190
CSS declaration	89
CSS Editor	14
CSS layouts	87
CSS selectors	89
CSSEdit	16

d

Data loading	98
DeBabelizer LE	187
design styles	26
Digimarc	107

Director	34, 80, 114
distance learning	11
DIV	87
Djing	10
Doctorow, Cory	39
Drag (ActionScript)	82, 202
Dreamweaver	14, 19, 21, 51, 56, 83-4, 87
DRM	39, 109
drop shadow	126
dummy text	22
duplicating	159
DVD menus	130
Dynamic Text	81

e

eEye	12
Electrotank	116
Eraser tool	91
extra features	54
extrusions	146

f

Facial features	160
Field labels	99
Filezilla	16
Fill Transform tool	195
Firefox	57, 108, 194
Fireworks	56, 70, 80, 90, 100, 102, 126, 159, 164, 168, 175, 188, 190, 194
Fireworks Developer Center	72
Flash comics	63
Flash Decompiler Lite	212
Flash Designer	214
Flash games	74, 80
Flash games	112, 116
Flash movies	118
Flash showcase	112
Flash video	33
Flashkit.com	98
FlashTV.com	64
Float	87

FontFlasher/FontLab	148	Illustrator	157, 168	masking	163
Fontographer	141	Image 100	38	Maya	37, 40
Fonts, free	104	ImageReady	175, 185	Microsoft Office	37
Football	75	Images, tileable	118	Microsoft Word	199
footer	57	importing	159, 161	monitor calibration	181
forms	54, 200	Include command	71	morejamsandwiches.com	152
four-kings.com	12	insert image and text	22	Movie Properties	77
frame rate	191	Interactive media	68, 152	MovieClip	72, 77, 81-3, 92, 95, 121, 166, 169
Frameforge 3D Studio	131	Interface elements	121	MP3 player	199
frames, exporting	144	Internet 2.0	32	music streaming	199
frames, stepping through	137	Internet Explorer	58, 108, 194	n	
FreeHand	157, 168	Internet radio	10	NAV	149
FrontPage	175	iTrike	13	nested HTML	108
FTP	14	iTunes	29, 32	Netscape	194
g		j		Nightmare scenarios	42-45
Games programming	70-83	Japanese characters	98	Notepad	194
Getty Images	38	Java	90, 176	NVU	15
GIF	20, 109, 135, 179, 185	Java debugging	191	o	
GIF animation	186, 193	JavaScript	108, 174, 194, 200	Object oriented Programming	170, 195
GIMP	187	JavaScript rollovers	23	Onion skin	161
Global Grooving	10	JPEG	94, 179, 185	Online fonts	103
GoLive	175	k		online inspiration	28
Google	12	Keyboard mapping	146	online publishing	83
graphics, Web	184	Keyframes	158	Optimize!	137
guestbook	98	l		Oval tool	141
guides and rules	157	Layers	139	p	
GUIs, interactive	126	Layers, arranging	142	page properties	21
h		Legal reselling	15	Pagespinner	15
hand-drawn	29	Lego	115	Paint Shop Pro 9	187
Hi-Res	68	lens flare, faking	188	Paintbrush tool	136
Hittless	65	LightWave 3D	75	panoramic images	90
HTML	21, 56, 101-2, 174, 195	lip-synching	160	Passwords	94
HTML editor	14	Looped animation	119	PDF	195
HTML template	104	Lview Pro	187	Pen and paper	132, 152
HTML, nested	108	m		Pencil tool	140
i		mailto	195	Perl	176
Icon font	139	mask layer	72	Photo album	94
I-D Media	130			Photoshop	20, 109, 118, 130, 135, 139, 145, 175 >



Index continued...

Photoshop actions	146	SharedObject	126	V	
Photoshop Actions	186	Shockwave	26, 34, 70	vecta3d.com	137
Photoshop CS	107	shoot-'em-ups	80	vector tools	80
Photoshop CS	165	Site layout	102	Video, rotoscoping	135
PHP	174, 195	Skelegons	77	videoLAN	108
plagiarism	27	Sketchbook Pro	131	virtual tour	90
planetofthedrums.com	148	Soulbath	69	W	
PNG	57, 179, 185	Sparrowbrain	156	W3C	56
pocketlaboratories.com	152	stack	87	Wacom tablet	131
polygons	76	standards	36	Watermark	107
pop-up menus	196	status bar messages	194	Watermark	182
Poser 4	75	stock images	38, 104	WaxPraxis	72, 164
position images	23	Storyboard lite	131	Web graphics	179, 184
PostScript	142	Storyboard Quick/Artist	131	Webcasting	13
Powerbullet Presenter	216	storyboarding	130, 218	Web-safe colours	180
prototype	72	streaming music	199	Windows Media Player	24
punch command	70	structure, games	77		
q		Studio MX Games design	70-83	X	
Quicktime	34	Style dialog	89	XHTML	58
r		Subselection tool	198	XML	96, 148, 166, 202
rasterize	150	SVG	185	XMP	107
RealPlayer	34, 90	swf	21, 125, 158, 212	Z	
Realviz StoryViz	131	t		Zend	176
retouching	183	Tables and CSS	53-54	Zoolook	62
RGB	169, 187	Templates	56, 124		
RGB	187	testing	35		
RIA (Rich Internet Application)	126	The GIMP	187		
rights management	38	thumbnails	183		
rotoscoping	135	tileable images	118		
Royalty-free images	38, 118	tomorrow's internet	13		
s		Topstyle Pro	16		
Safari	194	transition	144		
Scorsese, Martin	133	Transmit	16		
scroll bar	202	TrueType	142		
scrolling text	196	tweening	156, 163		
SGML	174	typography	26, 143, 148		
Shadow effect	70	u			
		upload	105		
		usability	24		

BOOST YOUR SKILLS WITH **FLASH**

Everything you need to create compelling content

→ GETTING STARTED

Work smarter in just a few minutes with our unmissable multimedia know-how

→ RETURN OF THE CODE

Learn how ActionScripts and the command line can get you spectacular final results

→ INTEGRATE YOUR WORK

Find out how Flash interacts with the other elements of Macromedia Studio MX 2004

→ FANTASTIC FLASH PROJECTS

Use Flash to build your own countdown timer, driving game, shoot-'em-up and more!

→ TRICKS OF THE TRADE

Web professionals reveal their favourite cutting-edge techniques and shortcuts

→ AWESOME ANIMATION

All the tutorials and software you need to storyboard and animate characters onscreen

→ ON YOUR BONUS CD-ROM...

- Macromedia Studio MX 2004 trial version
- Flash Decompiler Lite full program
- All the files you need to follow our walkthroughs
- Flash tools, Macromedia players and more!



For complete cover disc contents, see page 210



PRICE £14.99 NET UK

Published by Future Publishing Limited
30 Monmouth Street, Bath BA1 2BW, United Kingdom
Telephone +44 (0) 1225 442244
www.futurenet.co.uk

All content © 2005 Future Publishing Ltd.